Fall 12-1-2008

Project Management Tool

Karteek Boyanapalli
Dakota State University

PROJECT MANAGEMENT TOOL

A graduate project submitted to Dakota State University in partial fulfillment of the requirements for the degree of

Master of Science

in

Information Systems

December, 2008

By
Karteek Boyanapalli

Project Committee:

Shan Ronghua
Stephen Krebsbach
Sreekanth Malladi

PROJECT APPROVAL FORM

We certify that we have read this project and that, in our opinion, it is satisfactory in scope and quality as a project for the degree of Master of Science in Information Systems.

Student Name: Karteek Boyanapalli

Master's Project Title: Project Management Tool

Faculty supervisor: Ronghua Shan Date: 12/5/08

Committee member: Stephen Krebsbach Date: 12/5/08

Committee member: M. Sreekanth Date: 12/02/08

ABSTRACT

In an organization there are many issues regarding project management. Managing the projects and the human resource is becoming difficult day by day. This 'Project Management Tool' is an attempt to solve those issues. It mainly helps project managers to monitor and control projects in an organization effectively.

Using this tool superiors can assign projects, modules, tasks to their sub-ordinates and can monitor the status. They can also send E-Mails to their sub-ordinates using the E-Mail generation feature. This project also enables the employees to submit their day-to-day work. This project generates three different reports namely project reports which gives details of all the projects, task reports which gives description of all the tasks in an organization and employee reports which gives details of all the employees.

This project is developed in asp.net with sql server as back end.

DECLARATION

I hereby certify that this project constitutes my own product, that where the language of others is set forth, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions or writings of another.

I declare that the project describes original work that has not previously been presented for the award of any other degree of any institution.

Signed,

_____

Karteek Boyanapalli

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

When we think of project management it mainly involves processes, which includes considerable data, measurements, analysis and communication. The need for project management was driven by businesses that realized the benefits of organizing work around projects and the critical need to communicate and co-ordinate work across departments and professions. Most of the processes involved in managing projects cannot be handled effectively without the aid of project management software. Hence, organizing for project management will include the selection of project management tools.

1.1 Problem Definition

In the Present Day System, tasks are being carried out manually in the organizations which is time consuming and tedious. It requires the coordination of many different people and completion of many tasks in a precise sequence that would take a great deal of time to manage manually. When we think of the maintenance of the Employee records manually, it is very difficult to maintain them. The user may not get the required details regarding the Employee, projects or tasks efficiently.

In this type of situation, it is convenient for the user when he has a tool that can be used to carry out the tedious tasks without spending lot of time and which maintains all the details of projects at one place.

1.2 Problem Solution

Using our Project Management Tool, we can solve the above problem by carrying out the work in the form of assignment of tasks from superiors to sub-ordinates in an efficient manner. Reports regarding the projects, employees and tasks can be retrieved in a consolidated manner. When it comes to the communication among the employees, Emails could be generated to the respective employees regarding any issues in the Organization that accounts for the effective communication between the superiors and their sub-ordinates.

1.3 Project Description

PMT is a tool used to manage and organize the resources in such a way that these resources deliver the work required to complete the project within the defined scope, time and cost constraints. It also helps the project manager to oversee the workings and development of an entire project ensuring that everything is co-ordinate and planning is at its optimum.

This project management tool can be used in any organization for effective controlling and monitoring of projects. The people who can use this tool are administrator, project manager, team leader, team member.

Administrator

The Administrator is a person having administrative or managerial authority in an organization.

In this tool he is responsible for maintaining the records of employees.

Project Manager

Project manager after the client interaction, accepts the projects and monitors them. Generally projects will be divided into modules and modules into tasks. Project manager assigns a module under a project to a particular team leader and keeps track of its status. For team leader to work on the module he needs group of team members. So, project manager assigns team members to team leaders. Project manager assigns tasks to his subordinates. At the end of the day project manager submits the work done by him through timesheet. Project manager can view reports to get overall description of projects.

Team Leader

The person who understands the ultimate project objectives and who guides the rest of the team down the path through clear vision setting and effective communication is team leader.

After team leader is being assigned a module, he divides the module into tasks, and assigns them to team members. At the end of the day team leader fills timesheet which keeps track of hours worked by him.

Team Member

Team member has to perform the detailed and defined tasks necessary to complete a specific portion of the overall project or program. The team member has limited visibility into strategic planning and the work necessary to develop project and program plans. Team member primary focus is on mastering his area of expertise and gaining a firm grasp on the task at hand in order to deliver effectively.

He can view all the tasks assigned to him by his superiors and works on those tasks, after the completion of the tasks he submits them by filling time sheet.

Modules

The basic modules involved in our project are

1. Login
2. Registration
3. Task Delegation
4. Timesheet
5. Reports

Login

The user can get access to this tool with the help of this module

Registration

When ever an employee joins the organization, the administrator has to register him to this tool with the help of this module.

Task Delegation

Using this module tasks can be delegated by superiors to the subordinates, also the status of the work can be continuously monitored.

Timesheet

At the end of the day every employee has to fill the timesheet which keeps track of actual hours worked by that employee.

Reports

In project management tool we are generating 3 different types of reports. They are employee reports, project reports and task reports. The over all description of projects can be viewed in project reports. The details of employees can be viewed in employee reports. The total

number of tasks which are carried out by particular employee in the organization can be viewed in task reports.

1.4 Environment

The environment to run this project is systems with windows XP/windows 2000 operating system, which are connected through LAN.

1.5 Deliverables

This tool is basically an intranet application.

1.6 Software and Hardware Requirements

1.6.1 Software Requirements

Platform: Dot Net 2003

Database: Microsoft sql server 2000

Web Server: IIS (Internet Information Services)

Operating System: Windows XP, Windows 2000

1.6.2 Hardware Requirements

Processor: Pentium 3, Pentium 4, Celeron

RAM: 256MB or higher

1.7 Success Criteria

This project can be successfully used by organizations for effective task delegation and monitoring of projects.

CHAPTER 2

REQUIREMENT ANALYSIS

2.1 Understanding Concept

2.1.1Understanding Intranet

An intranet is a private computer network based on the communication standards of the Internet. It is smaller version of the Internet that only the members of an organization can see. Companies can create, within their walls, a manageable, secure version of the World Wide Web.

Intranet Features

1. Uses TCP/IP (Transmission Control Protocol/Internet protocol) for both wide-area and local-area transport of information.
2. Uses HTML (hypertext markup language), SMTP (Simple mail transport protocol) and other open Internet-based standards as the means of moving information from clients to servers.
3. It is completely owned by the corporation and not accessible from the Internet-at-large by the general public.

Why we use intranet

1. The Intranet is built on a technology standard that everyone uses. It is less expensive.
2. The tools to build an Intranet are the same that you use to build a Web page.
3. An Intranet provides opportunities for multimedia that internal corporate networks often do not provide.
4. Intranet is available 24 hours day like any other network.
5. The ability to hyperlink on an Intranet provides fast information access that other networks do not offer.

2.1.2 Why we developed intranet application

Intranets complement or substitute most internal employee communications in use. An Intranet allows faster communication, wider distribution and greater efficiency than many

printed and electronic media. As Project Management Tool involves interactions among employees we developed it as an intranet application.

2.2 Understanding ASP.NET

ASP.NET is a programming framework used to create enterprise-class Web Applications. These applications are accessible on a global basis leading to efficient information management.

The .NET Framework was introduced with a vision to create globally distributed software with Internet functionality and interoperability. Developing Internet applications with the .NET Framework is very easy. ASP.NET is built into this framework. We can create ASP.NET applications using any of the built-in languages.

ASP.NET uses the Common Language Runtime (CLR) provided by the .NET Framework. This CLR manages execution of the code we write. ASP.NET code is a compiled CLR code instead of interpreted code .CLR also allows objects written in different languages to interact with each other. The CLR makes development of Web applications simple.

ASP.NET is purely server-side technology. ASP.NET code executes on the server before it is sent to the browser.

Web Server (IIS):

Web servers run special software to support mail exchange, FTP, HTTP access, and everything else clients expect in order to access the web content. In order to use ASP.NET, your computer needs to act like a web server. In fact, while you are testing an ASP.NET application your development computer will work exactly the same way as it would over an internet connection to a remote client. When you test a page, you will actually access the page through IIS (internet information server) and retrieve the final HTML through an HTTP transfer.

It requires one to have the IIS software on our computer in order to use ASP.NET.IIS is included with Windows 2000,Windows XP ,and Windows NT, but it is included as an optional component and is not installed by default. To create ASP.NET programs successfully ,one need to make sure IIS is installed, preferably before you install the .NET framework or Visual Studio .NET.

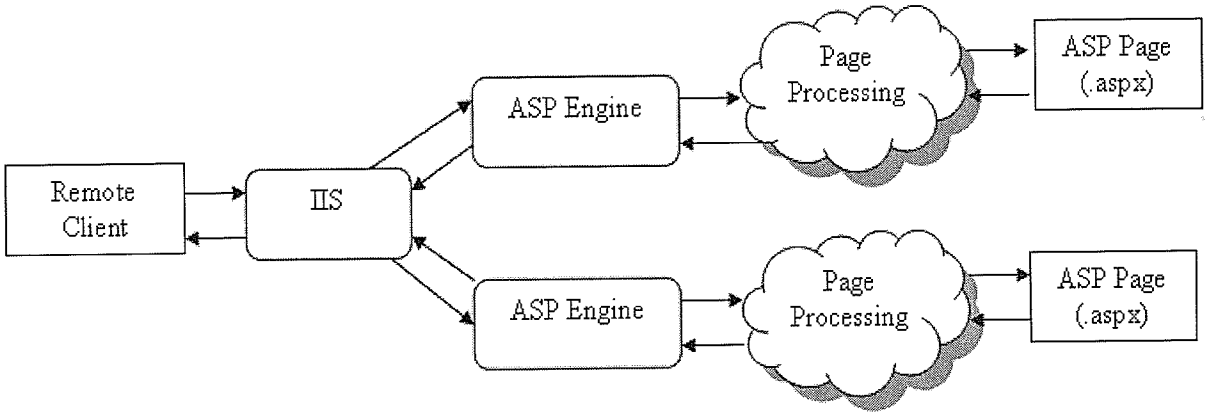

Figure 1. ASP.Net page compilation

2.2.1 ADO.net in ASP.NET

ADO.NET (Activex Data Objects) is a set of computer software components that can be used by programmers to access data and data services.ADO.net allows users to connect to database in two ways.

1) Connection oriented model

2) Disconnected model

Connection oriented model

In this type of connectivity, connection is required between application and database for every manipulation.

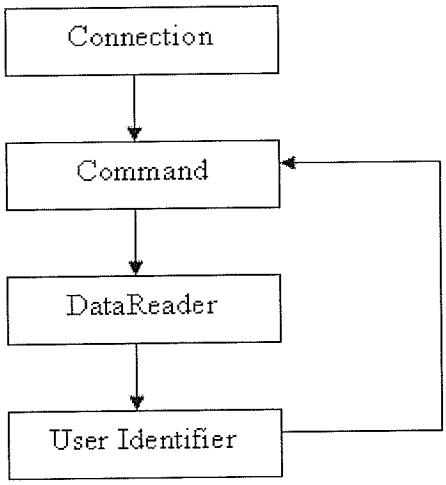

Figure 2. Connection oriented model

Disconnected model

In this type of connectivity connection is required between application and database only when passing queries.

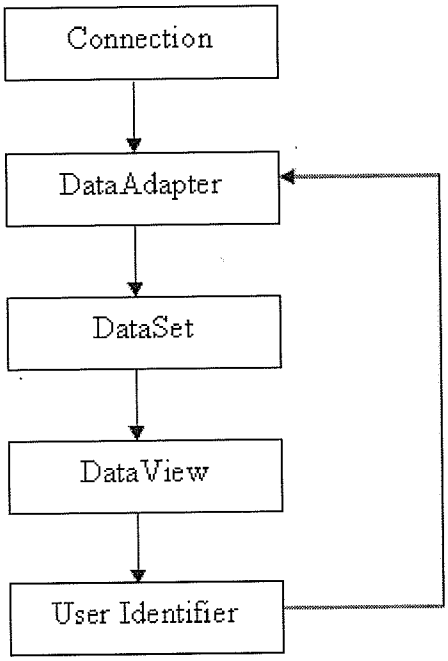

Figure 3. Disconnected model

Connection

This class is used to establish physical connection between application and database.

Dim obj as xxxconnection

xxxconnection=new xxxconnection (servername, databaename, userid, password)

Command

Command class is used to execute queries.

Dim objname as xxxcommand

Objname=new xxxcomand ()

Data Adapter

It is a collection of commands which acts as bridge between application and database that helps to transfer data.

Dataset

Dataset is a cache memory which is capable of storing any number of tables from same or different database.

2.2.2 Session management in ASP.NET:

Session object can be used for storing session-specific information that needs to be maintained between server round trips and between requests for pages. Session object is per-client basis, which means different clients generate different session object. The ideal data to store in session-state variables is short-lived, sensitive data that is specific to an individual session.

Storing information in the session:

Session ["myname"] ="ABC";

Retrieving information from the session:

myname=Session ["myname"];

Session state is global to entire application for the current user. Session state can be lost in several ways:

1. If the user closes and restarts the browser.
2. If the session times out due to inactivity.
3. If the programmer ends the session in code.
4. If the user accesses the same page through a different browser window.

2.2.3 Sending Mail in ASP.NET:

ASP.net provides a built-in option of sending mails. The mailing feature uses the built-in SMTP service included with IIS.

The namespace appropriate for working with mails is System.Web.Mail.

One needs to create a new mail message object, and set properties to identify the recipient, the priority, the subject, and the message itself.

2.3 Architecture Diagram

Project Management Tool follows 3-tier architecture model. A 3-tier application is a program which is organized into three major disjunctive tiers. These tiers are

- Presentation Tier (Front end)
- Logical Tier (Middleware)
- Data Tier (Backend).

Each layer can be deployed in geographically separated computers in a network. They can also be deployed on physically separated machines. The characteristic of the tier communication is that the tiers will communicate only to their adjacent neighbours. For an example, The Presentation Tier will interact directly with the Business Tier and not directly with Data Access or Data Tiers.

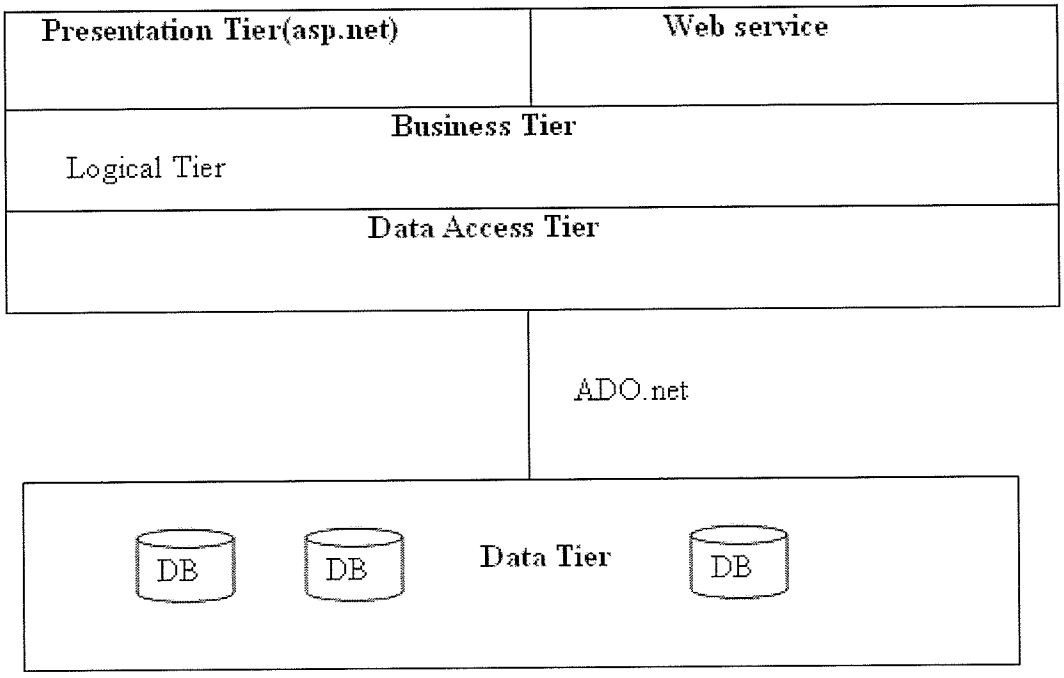

Figure 4. Typical three tier architecture

Data Tier

This Tier is responsible for retrieving, storing and updating Information .Therefore this tier can be ideally represented through a commercial database. We consider stored procedures as a

part of the Data Tier. Usage of stored procedures increases the performance and code transparency of an application.

Business Tier

This sub tier contains classes to calculate aggregated values and this tier does not know about any GUI controls and how to access databases. The classes of Data Access Tier will supply the needy information from the databases to this sub tier.

Data Access Tier

This tier acts as an interface to Data Tier. This tier knows how to (from which database) retrieve and store information.

Presentation Tier

This Tier is responsible for communication with the users and web service consumers and it will use objects from Business Layer to response GUI raised events.

2.4 Various Modules

Before going to the modules involved in our project let us understand the need for modularization.

The Modules involved in PMT

1) Login.
2) Registration.
3) Task Delegation.
4) Time Sheets.
5) Reports.

Login

The Login Module is mainly used by the employees to get access to the tool. Here, based on the unique Login ID and Password, users such as Administrators, Project Managers, Team Leaders and Team Members can manage processes specific to their business. When the employee logs in session is created and is maintained till he logs out. And later it will be destroyed, restricting the illegal access to the tool by other employees.

Registration

Registration is a process of registering the employees of a company. Whenever a new employee joins the organization administrator enters all the details like employee First name, Last name, Date of joining, Designation, Employee Type, Official E-Mail ID, Alternate E-Mail ID, Employee Phone Number, Experience, Areas Worked etc. After entering the details, the Module will automatically register the employee to the tool. It generates username and password randomly and sends the same to the employee mail box along with URL (uniform resource locator).

For the first time, employee logs in with the user ID and password assigned to him. As the password is randomly generated employee is given an option to change his password.

Task Delegation:

Task Delegation can be defined as the process where 'n' no of tasks would be delegated to the subordinates by their superiors.

Generally projects will be divided into modules and modules into tasks. After a project is signed by the organization it is assigned to project manager. Project manager estimates the cost and time and adds the projects to the tool. Then, projects will be divided into modules and will be delegated to respective team leaders. Also, a team of members are assigned to team leader to process the module. Later, team leaders divide modules into tasks and assign them to their team members.

At any point of time, team member can view the tasks assigned to him. A mail is also generated to the person intimating about assigned tasks.

The tasks assigned by superiors will be continuously monitored. If any assigned task is falling behind the schedule alerts can be sent to the respective persons through E-Mail.

Time sheet

Time sheet is a form that employee fills out to indicate how much time they have spent performing a job.

At the end of the day all the employees of the organisation need to fill the timesheet.

Timesheet are filled:

A) To submit the work done by a person.

B) To submit the actual hours spent on a task.

C) To submit the status of the task.

Once, time sheet is filled by an employee for a task the reflections are made to the actual hours worked and status. Time sheet provides flexibility to user to add task to him self which can be a miscellaneous task.

Reports

The Overall statistics of the project can be viewed in reports.

In this tool we have 3 kinds of reports

- Project reports
- Employee reports
- Task reports

Project Reports

This project reports will help to communicate the status of the project to the staff. It also enables the people informed of the progress of the project regularly and to raise any issues for their attention .In the project reports, one can view the details like total no. of modules, tasks, status of the project, number of due date extensions, assigned users. One can also view the detailed description of all the modules and tasks involved in each project.

In addition to above features for the retrieval of the data in the desired fashion we go for filters. The filters provided here are

a) Based on type of the project (Billable/Non Billable).

b) Status of the project.

c) Based on assigned to.

For instance, if we want list of all the projects of type 'Billable', we can get this by applying filter. Apart from this we also provided search criteria that enable the users to get the relevant information they want.

The search criterion is:

a) Searching pending projects.

b) Searching projects by Date.

c) Searching projects by Client name.

d) Searching projects by project name.

Task Reports

Task reports give overall tasks carried out in the organisation by all employees. Here also we applied filters based on projects, status and assigned to. If one wants to know all the tasks carried out by a person this can be obtained by applying the 'Assigned to' filter.

Employee Reports

Employee reports give the details of all the employees in the organization. i.e currently working and ex-employees.

Project Manager and Administrator can view all the employee details but team leader is restricted to view only his sub-ordinate's details. By seeing the employee reports one can view the total no. of employees. To get specific information, here also we can apply filters. The filters provided here are based on designation and experience. Here, we can also search people based on their specialization.

CHAPTER 3

DESIGN

3.1 UML as design tool

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing Object oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML, helps project teams communicate, explore potential designs, and validate the architectural design of the software.

3.2 Types of UML diagrams used in this project

Each UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction. Some UML diagrams include:

Use Case Diagram

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors.

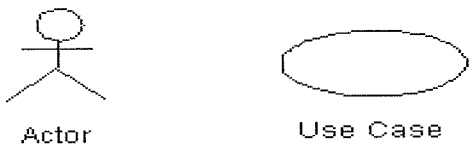

Figure 5. Components of Use Case Diagram

An actor represents a user or another system that will interact with the system you are modelling. A use case is an external view of the system that represents some action the user might perform in order to complete a task.

Class Diagram :

Class diagram models class structure and contents using design elements such as classes, packages and objects. It also displays relationships such as containment, inheritance, associations and others.

Classes are composed of three things: a name, attributes, and operations. Below is an example of a class.

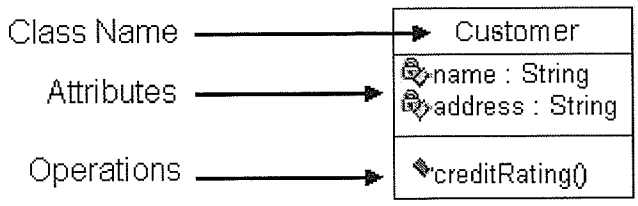

Figure 6. Elements of Class Diagram

Activity Diagram

Activity diagram is a special state diagram where most of the states are action states and most of the transitions are triggered by completion of the actions in the source states. This diagram focuses on flows driven by internal processing. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel.

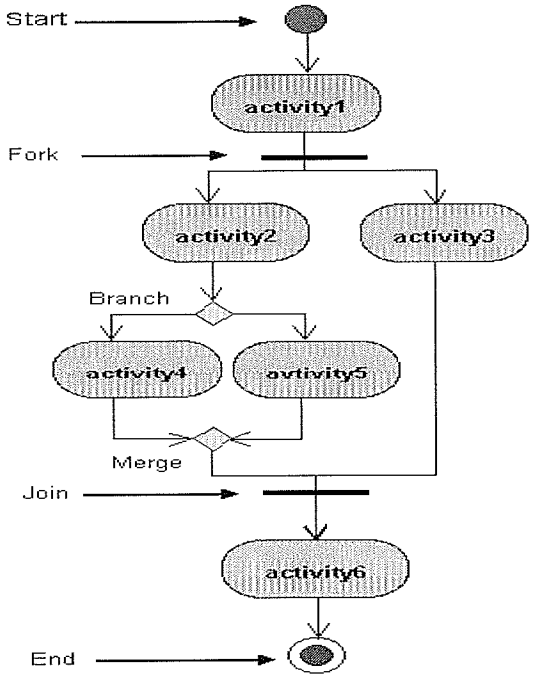

Figure 7. Activity Diagram

3.3 Various UML Diagrams

3.3.1 Use case diagrams

Use case diagram for administrator

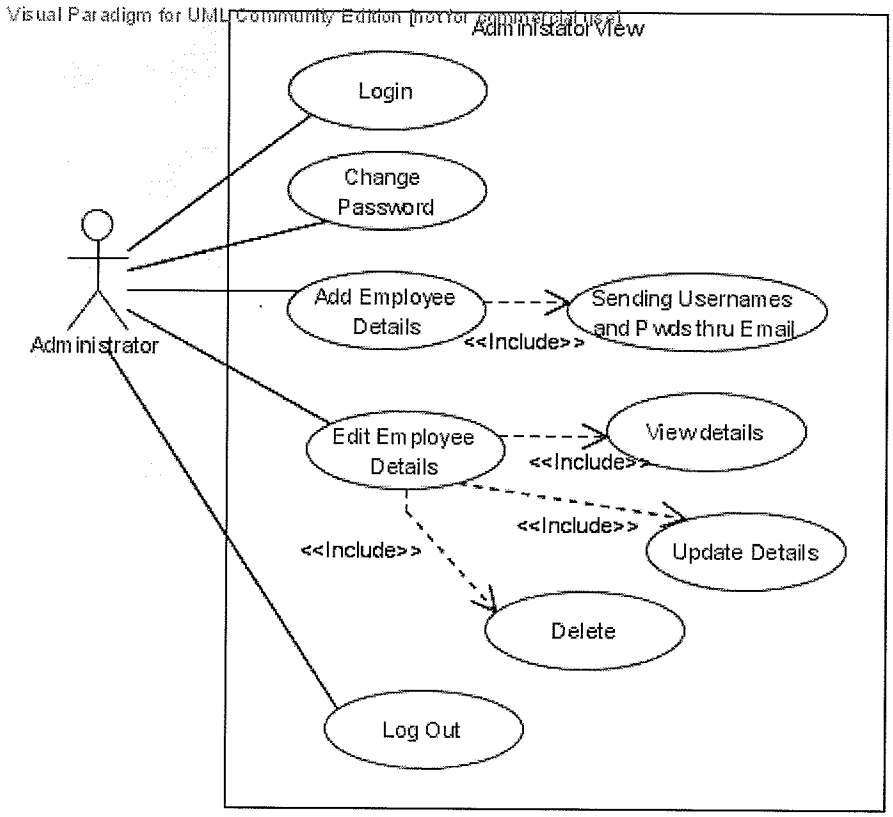

Figure 8. Use Case Diagram for administrator

Use case diagram for project manager

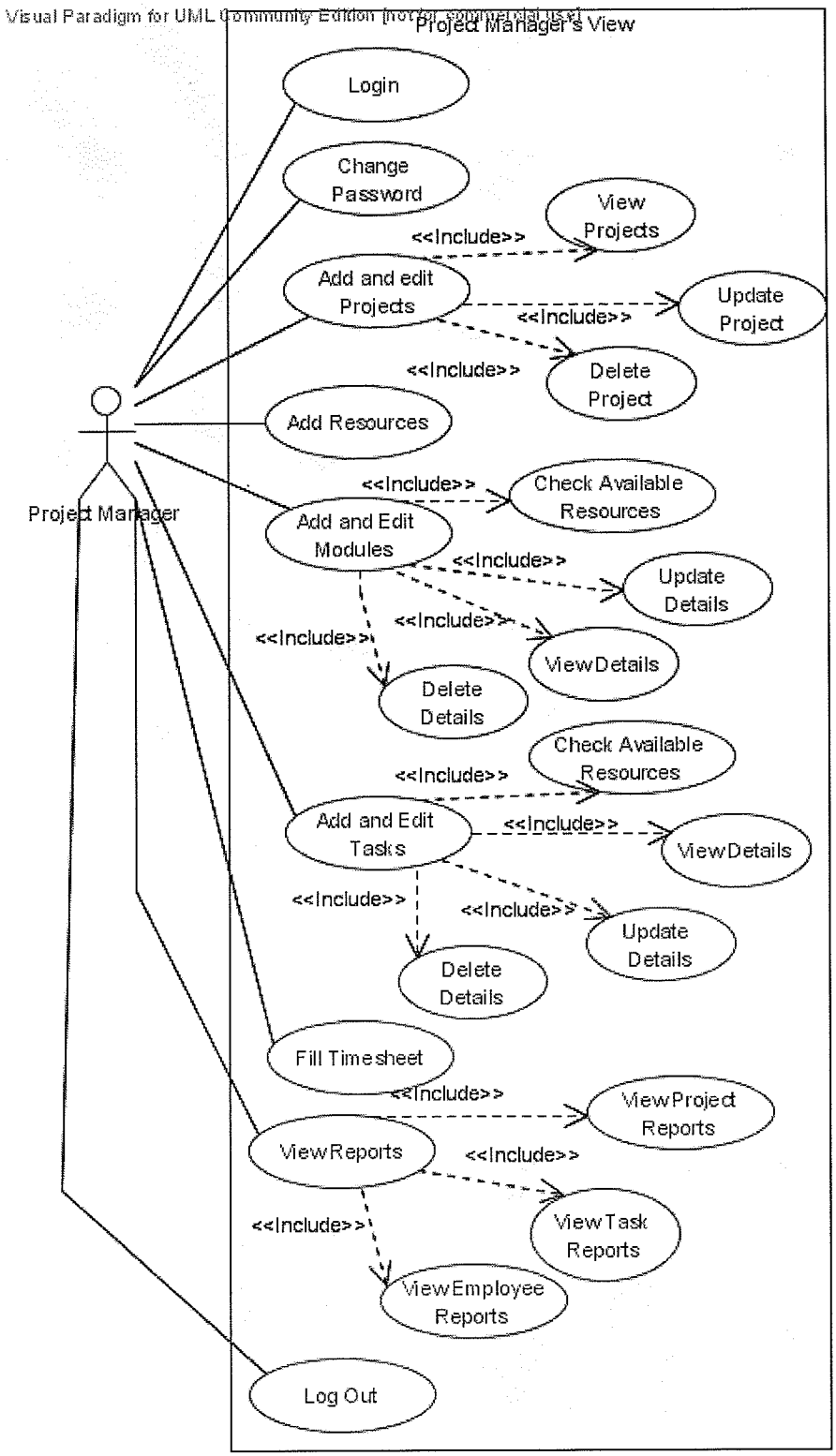

Figure 9. Use Case Diagram for project manager

Use case diagram for team leader

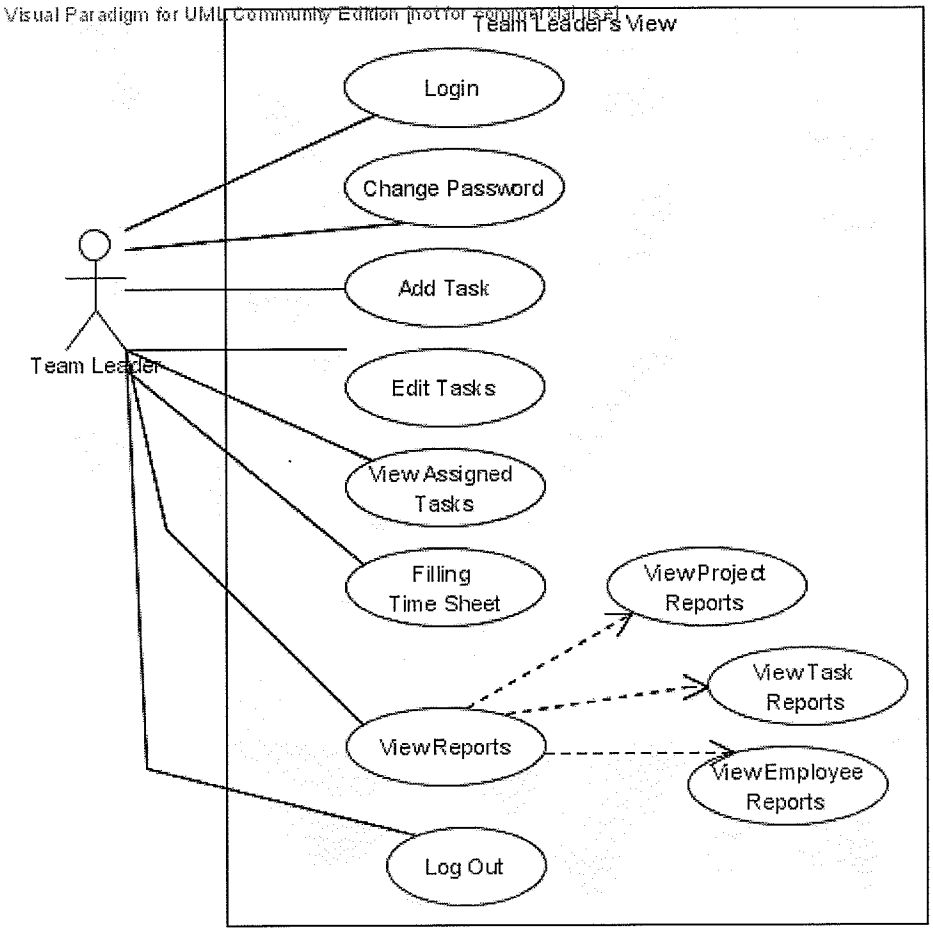

Figure 10. Use Case Diagram for team leader

Use case diagram for team member

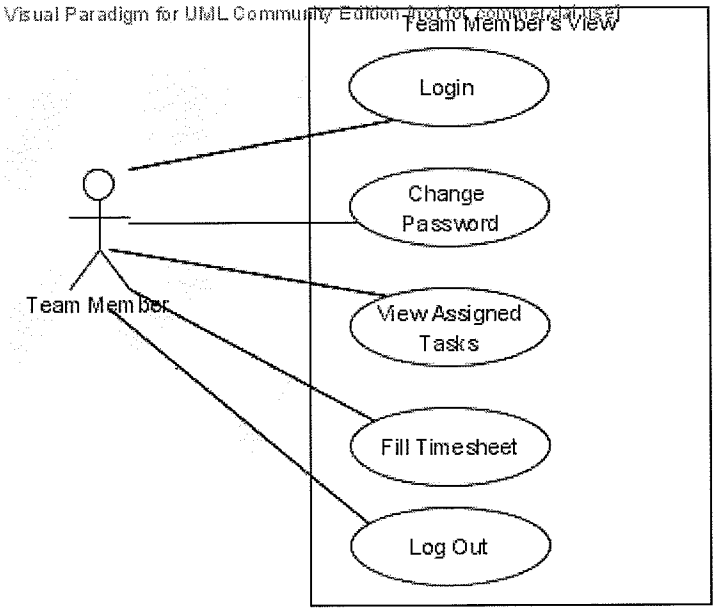

Figure 11. Use Case Diagram for team member

3.3.2 Class Diagram

Class Diagram for project management tool

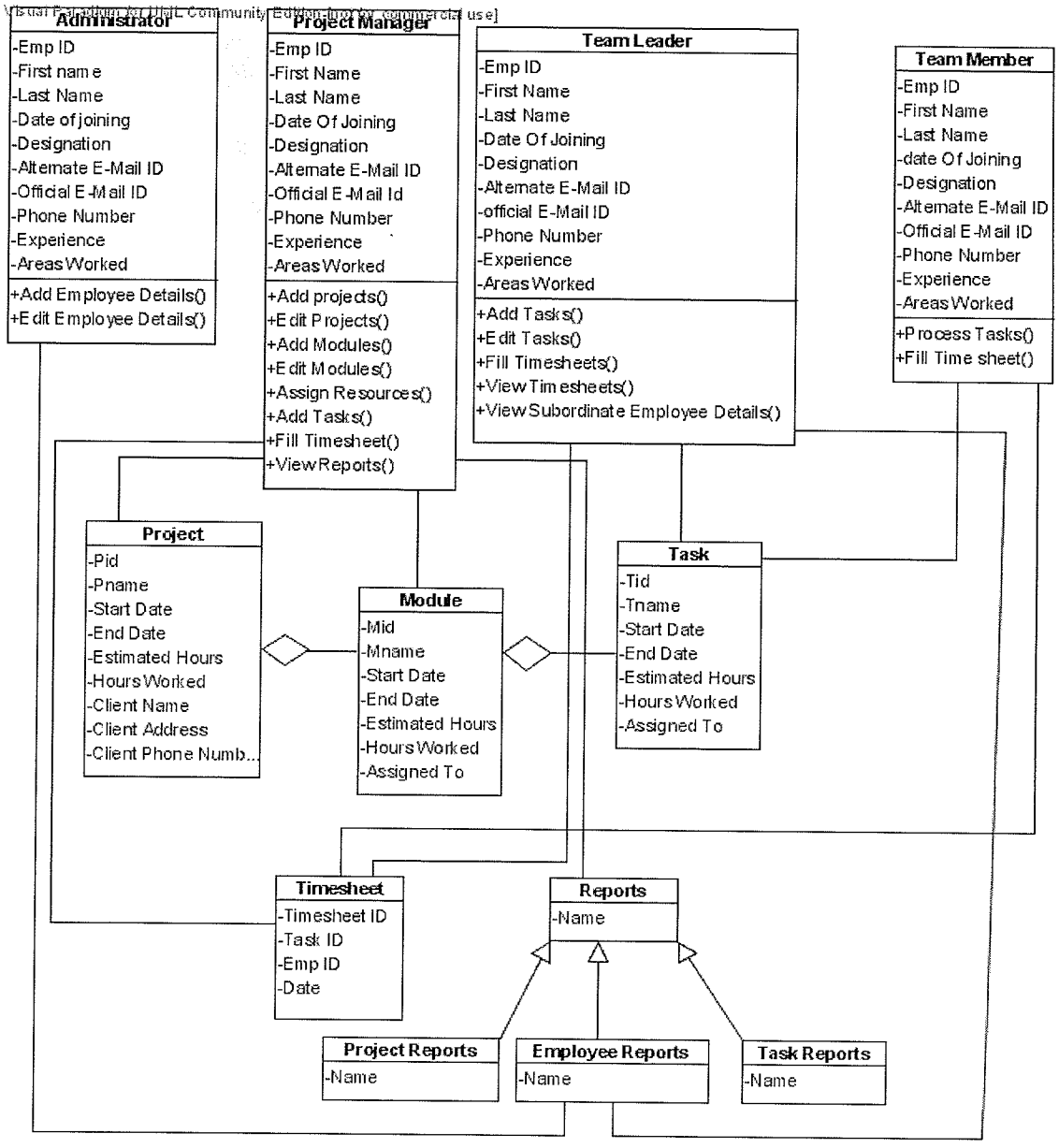

Figure 12. Class Diagram for project management tool

3.3.3 Activity Diagrams

Activity Diagram for Login module

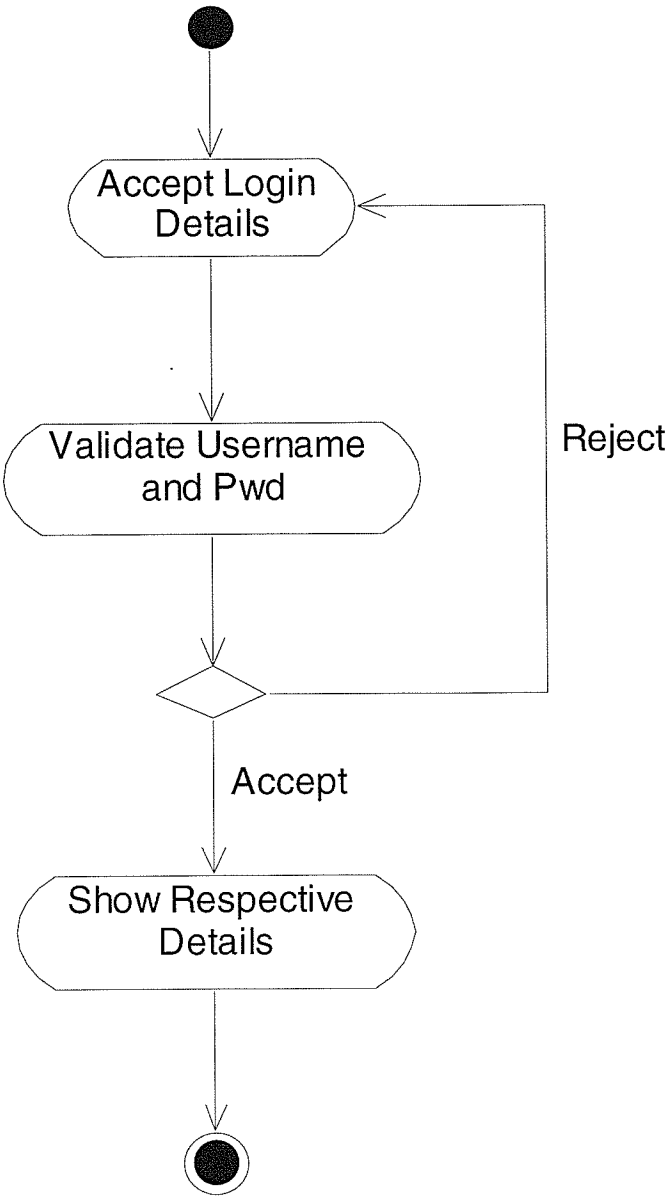

Figure 13. Activity Diagram for Login module

Activity Diagram for Registration module

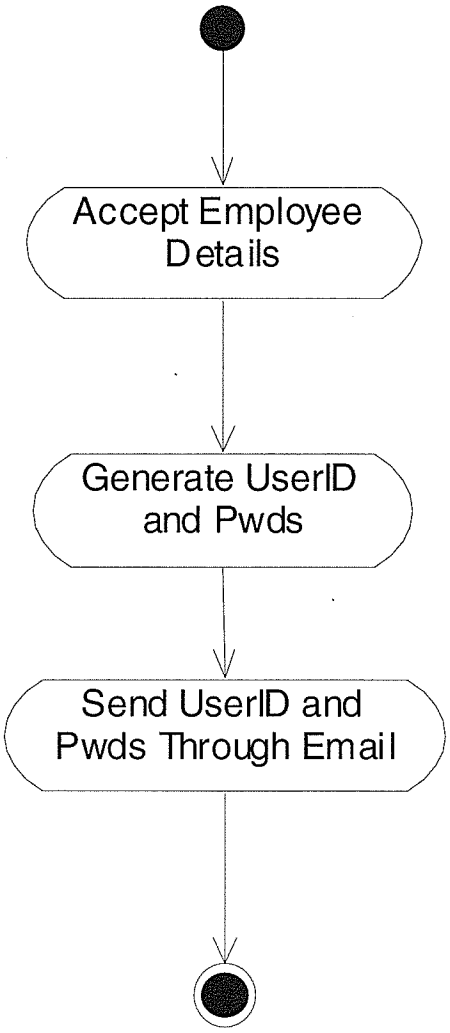

Figure 14. Activity Diagram for Registration module

Activity Diagram for Task delegation module

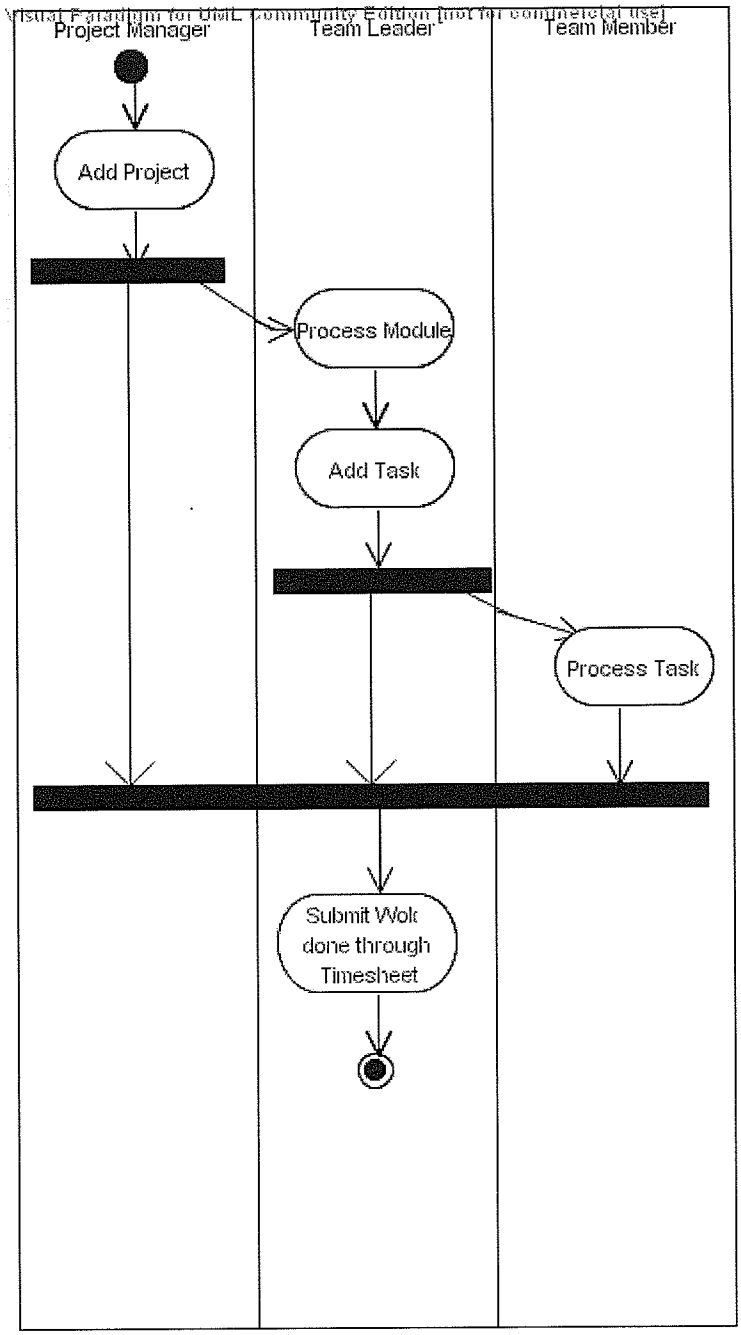

Figure 15. Activity Diagram for Task delegation module

CHAPTER 4

IMPLEMENTATION

4.1 Creation of tables

4.1.1 What is Database?

Database is a collection of information that has been systematically organized for easy access and analysis.

Database Table

A table can be the basic component of a relational database. It has number of rows and columns, and can usually be manipulated using SQL commands.

The basic key constraints we apply for tables are:

1) Primary key
2) Foreign key

Primary key: A primary key is a column which uniquely identifies rows in the table. A primary key value cannot be null.

Foreign key: A foreign key is a column in a table that does not uniquely identify rows in a table but is used as a link to matching columns in other tables.

4.1.2Accessing the database

The information stored in database can be accessed either by using SQL commands or Stored Procedures.

Various SQL commands used are:

1) Insert
2) Update
3) Delete
4) Select

Insert

Insert command is used to insert rows into tables.

Insert into <table name> values (<value1, value 2.....value n>)

Update

Update command is used to update rows in the tables.

Update <table name> set <value1=new value1, value 2=new value 2.....value n=new value n> where <condition>

Delete

Delete command is used to delete rows from the table.

Delete from <table name> where <condition>

Select

Select command is used to select rows from the table.

Select from <table name> where <condition>

Stored procedure

Stored Procedures are precompiled database queries that improve security, efficiency, usability of client server applications.

4.1.3 Tables in the Project

Our project has 13 database tables. They are:

1) Employee Designation table:-

Table 1.Employee Designation table

Field	Datatype	Size
iempid	Integer	4 bytes
idesignationid	Integer	4 bytes

Primary Key: idesignationid
Foreign Key: iempid

Whenever a new employee is registered then a unique designation id is given and is stored in the above table.

2) Designation table:-

Table 2.Designation table

Field	Datatype	Size
Idesignationid	Integer	4 bytes
Vdesignation	Varchar	255 characters

Primary Key: idesignationid

The employee at the time of registration is also given a designation id and type of designation that is being stored in this table.

3) Login Details table:-

Table 3.Login Details table

Field	Datatype	Size
iloginid	Integer	4 bytes
vloginname	varchar	255 characters
vpassword	varchar	255 characters
iempid	Integer	4 bytes
vemptype	varchar	255 characters

Primary Key: iloginid
Foreign Key: iempid

When the employee logs on to the tool, his login name and password are being stored in this table.

4) Employee type table:-

Table 4.Employee type table

Field	Datatype	Size
iemptypeid	Integer	4 bytes
vemptype	Integer	4 bytes

Primary Key: iemptypeid

At the registration of a particular employee, the emptype and the emptypeid are being stored in the above table.

5) Employee Details table:-

Table 5.Employee Details table

Field	Datatype	Size
iempid	Integer	4 bytes
vempfirstname	varchar	255 characters
vemplastname	varchar	255 characters
vempaddress	varchar	255 characters
voffemail	varchar	255 characters
valtemail	varchar	255 characters
vphone	varchar	255 characters
sddateofjoin	smalldatetime	4 bytes
barchive	bit	1unit
rexperience	Real	8 bytes

Primary Key: iempid

At the time of registration, all the employee details are being stored in the above table.

6) Project table:-

Table 6.Project Details table

Field	Datatype	Size
Ipid	Integer	4 bytes
Vpname	Varchar	255characters
sdstartdate	Smalldatetime	4 bytes
Sdenddate	Smalldatetime	4 bytes
Vcomments	Varchar	255characters
Vtype	Varchar	255characters
Istatusid	Integer	4 bytes
Iestimatedhours	Integer	4 bytes
Vclientname	Varchar	255characters
Vclientaddress	Varchar	255characters
Vcleintemailid	Varchar	255characters
Vclientphone	Varchar	255characters

Primary Key: ipid

When the Project Manager starts adding the projects, all the details regarding the project are being stored in the above table.

7) Module table:-

Table 7.Module Details table

Field	Datatype	Size
imid	Integer	4 bytes
ipid	Integer	4 bytes
Vmname	Varchar	255characters
iassignedto	Integer	4 bytes
Sdstartdate	Smalldatetime	4 bytes
Sdenddate	Smalldatetime	4 bytes
iactualhours	Integer	4 bytes
iestimatedhours	Integer	4 bytes
istatusid	Integer	4 bytes
Barchive	Bit	1

Primary Key: imid
Foreign Key: ipid

When the project Manager adds the module, all the above details are being stored in the above table.

8) Status table:-

Table 8.Status table

Field	Datatype	Size
istatusid	Integer	4 bytes
cstatusname	character	1byte

Primary Key: istatusid

In the above Table the Status of the Project, module or the task i.e. either created, in progress or finished is being stored in the above table.

9) Task table:-

Table 9.Task table

Field	Datatype	Size
imid	Integer	4 bytes
itaskid	Integer	4 bytes
vtaskname	Varchar	255characters
iassignedto	Integer	4 bytes
iassignedby	Integer	4 bytes
sdstartdate	Smalldatetime	4 bytes
sdenddate	Smalldatetime	4 bytes
iestimatedlhours	Integer	4 bytes
ihoursworked	Integer	4 bytes
istatusid	Integer	4 bytes
barchive	Bit	1
ipid	Integer	4 bytes

Primary Key: itaskid
Foreign Key: imid, ipid

Whenever the Project Manager or the Team Leader assigns the tasks, the respective fields are being stored.

10) Resource table:-

Table 10.Resource table

Field	Datatype	Size
Iresourceid	Integer	4 bytes
Ipid	Integer	4 bytes
Iempid	Integer	4 bytes
Iassignedempid	Integer	4 bytes

Primary Key: iresourceid
Foreign Key: ipid, iempid, iassignedempid

When the Project Manager or the Team Leader can assign the particular resource for a particular project, then the above details are being stored in the table.

11) Miscellaneous Task Table:-

Table 11.Miscellaneous Task

Field	Datatype	Size
iempid	Integer	4 bytes
vtaskname	Integer	4 bytes
sddate	smalldatetime	4 bytes
ihoursworked	Integer	4 bytes
ctype	Character	1byte

Foreign Key: iempid

In case if there are any Miscellaneous Tasks details they are being stored in the above table.

12) ChangeRequest table:-

Table 12.Change Request table

Field	Datatype	Size
ipid	Integer	4 bytes
sdenddate	Smalldatetime	4 bytes
sdlastmodified	Smalldatetime	4 bytes
iestimatedhours	Integer	4 bytes
Icost	Integer	4 bytes

Foreign Key: ipid

If at all the Project Manager wants to extend the due Date or Estimated hours or cost of the Particular Project, then the respective fields would be updated and stored in the above table.

13) Timesheet table:-

Table 13.Timesheet table

Field	Datatype	Size
Iempid	Integer	4 bytes
Itaskid	Integer	4 bytes
Sddate	smalldatetime	4 bytes
Ihours	Integer	4 bytes
Ctype	Character	1byte

Foreign Key: iempid, itaskid

At the end of the Day, when each employee starts filling the Timesheet, the respective fields would be stored in the above table.

4.2 Implementation of Coding

We used vb.net to code our project.

Sample code to add employee details and generate mail to users

```
Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Dim rand As New System.Random (System.DateTime.Now.Millisecond)
    Dim str () As String = Split (txtoffemail.Text, "@", -1)
    password = str (0) & Math.Floor ((rand.Next / 100000))
    'Dim str2 As String = txt_offemail.Text
    'insertion
    If (insert_values_empdetails () = True) Then
      If (insert_values_logindetails () = True) Then
        For desigcount = 0 To lst_designame.Items.Count - 1
          If lst_designame.Items (desigcount).Selected Then
            Dimdesigid=get_desigid (lst_designame.Items (desigcount).Value.ToString)
            insert_values_empdesignation (desigid)
```

```
            End If
          Next
        Else
          " lblmsg.Text = "Error in inserting record"
        End If
      End If
      email_generation ()

End Sub
#Region "insertion - user code from here"

'insert the values in the table empdetails ()


Private Function insert_values_empdetails () As Boolean
      date1 = Request.Form ("theDate1")
      Try
        cmd = New SqlCommand
        cmd.CommandType = CommandType.StoredProcedure
        cmd.Connection = cn
        cmd.CommandText = "proc_empdetails"
        cmd.Parameters.Add ("@firstname", txtfname.Text)
        cmd.Parameters.Add ("@lastname", txtlname.Text)
        cmd.Parameters.Add ("@address", txtaddr.Text)
        cmd.Parameters.Add ("@offemail", txtoffemail.Text)
        cmd.Parameters.Add ("@altemail", txtaltemail.Text)
        cmd.Parameters.Add ("@phone", CInt (txtpno.Text))
        cmd.Parameters.Add ("@dateofjoin", date1)
        cmd.Parameters.Add ("@archive", 1)
        cmd.Parameters.Add ("@experience", txtexp.Text)
        cmd.Parameters.Add ("@areasworked", txtareas.Text)
         If cn.State = ConnectionState.Open Then
           cn.Close ()
```

```
        End If
        cn.Open ()
        If cmd.ExecuteNonQuery () > 0 Then
          lbl_msg.Text = "Details EnteredSucessfully"
          Return True
        End If
      Catch ex As Exception
        Return False
      Finally
        cn.Close ()
      End Try
    End Function

'insertion into login details table

Private Function insert_values_logindetails () As Boolean
      Dim  emptypeid  As Integer
get_emptypeid(ddl_emptype.SelectedItem.Value)
      Try
        cmd = New SqlCommand
        cmd.CommandType = CommandType.StoredProcedure
        cmd.Connection = cn
        cmd.CommandText = "proc_logindetails"
        cmd.Parameters.Add("@loginname",txtoffemail.Text)
        cmd.Parameters.Add("@password", password)
        cmd.Parameters.Add("@empid", get_eid())
        cmd.Parameters.Add("@emptypeid", emptypeid)
        If cn.State = ConnectionState.Open Then
          cn.Close()
        End If
        cn.Open()
```

```
        If cmd.ExecuteNonQuery() > 0 Then
          Return True
        End If
      Catch ex As Exception
        Return False
      Finally
        cn.Close()
      End Try
    End Function

    'insertion into empdesignation table

Private Function insert_values_empdesignation(ByVal empdesigid As String) As Boolean
    Dim empid As Integer = get_eid()
    If empid <> 0 Then
      Try
        cmd = New SqlCommand
        cmd.CommandType = CommandType.Text
        cmd.Connection = cn
        cmd.CommandText = "insert into
                    empdesignation values("
                  & empid & "," & empdesigid
                              & ")"
        If cn.State = ConnectionState.Open Then
          cn.Close()
        End If
        cn.Open()
        If cmd.ExecuteNonQuery() > 0 Then
          Return True
        End If
      Catch ex As Exception
```

```
            Return False
        Finally
            cn.Close()
        End Try
    Else
        Return False
    End If
End Function


'retrive empid from empdetails

Private Function get_eid() As Integer
    Dim eid As Integer
    Try
        cmdeid = New SqlCommand
        With cmdeid
            .CommandType = CommandType.Text
            .Connection = cn
            .CommandText = "select iempid from
                            empdetails"
        End With
        If cn.State = ConnectionState.Open Then
            cn.Close()
        End If
        cn.Open()
        dr = cmdeid.ExecuteReader()
        While (dr.Read())
            eid = CInt(dr("iempid"))
        End While
        Return eid
```

```
    Catch ex As Exception
        Return 0
    Finally
        cn.Close()
    End Try
End Function

Private Function get_desigid(ByVal designation As
                        String) As Integer
    Try
        cmd = New SqlCommand
        With cmd
            .CommandType = CommandType.Text
            .Connection = cn
            .CommandText = "select idesignationid from
                            designation where
                        vdesignation='" &
                            designation & "'"
        End With
        If cn.State = ConnectionState.Open Then
            cn.Close()
        End If
        cn.Open()
        Dim id As Integer = cmd.ExecuteScalar()
        Return id
    Catch ex As Exception
    Finally
        cn.Close ()
 End Try
End Function
```

```
'getting emptype id from emptype
  Private Function get_emptypeid (ByVal emptype As String)
                                  As Integer
    Try
      cmd = New SqlCommand
      With cmd
        .CommandType = CommandType.Text
        .Connection = cn
        .CommandText = "select iemptypeid from
                        emptype where
                      vemptype='" & emptype & "'"
      End With
      If cn.State = ConnectionState.Open Then
        cn.Close()
      End If
      cn.Open()
      Dim id As Integer = cmd.ExecuteScalar()
      Return id
    Catch ex As Exception
    Finally
      cn.Close()

    End Try
  End Function

Sub update_emptypecombo()
    Try
      cmd = New SqlCommand
      cmd.CommandType = CommandType.Text
      cmd.Connection = cn
      cmd.CommandText = "select vemptype from
```

```
                        emptype"
        cn.Open()
        dr = cmd.ExecuteReader
        While (dr.Read())
            ddl_emptype.Items.Add(dr(0))
        End While

    Catch ex As Exception
        MsgBox(ex.Message)
    Finally
        cn.Close()
        dr.Close()

    End Try

End Sub
Sub designation_listbox()
    Try
        cmd = New SqlCommand
        cmd.CommandType = CommandType.Text
        cmd.Connection = cn
        cmd.CommandText = "select vdesignation from
                        designation"
        cn.Open()
        dr = cmd.ExecuteReader
        While (dr.Read())
            lst_designame.Items.Add(dr(0))
        End While

    Catch ex As Exception
        MsgBox(ex.Message)
```

```
        Finally
            cn.Close()
            dr.Close()
        End Try
    End Sub
#End Region
Private Function email_generation()
        ' lblmsg.Text = password
        Dim obj As MailMessage
        obj = New MailMessage
        obj.From = getfrom()
        obj.To = txtaltemail.Text
        obj.Subject = "request for PMT registration"
        Dim str As String = "<a href='http://localhost/pmt/login.aspx'>click me</a>"
        obj.Body = "UserId: " & txtoffemail.Text & "Password: " & password & str & " ."
        obj.Priority = MailPriority.High
        obj.BodyFormat = MailFormat.Html
        Mail.SmtpMail.Send(obj)
    End Function
```

CHAPTER 5

SCREENS

Login Page

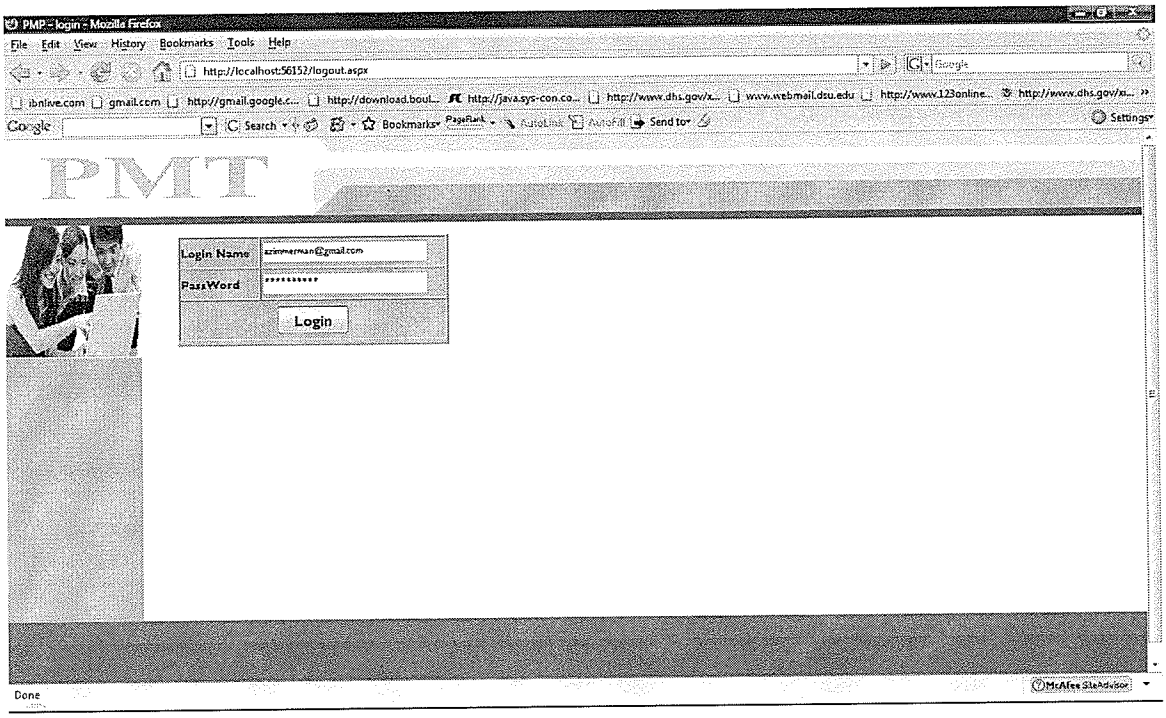

Figure 16. Login screen

Through Login screen employees access the project management tool. Login name and password will be used for authenticating employees. Once Authentication is done, based on the type of the Employee i.e. Administrator, Project Manager, Team Leader and Team member, the user will be directed to their respective modules.

Welcome page for Administrator

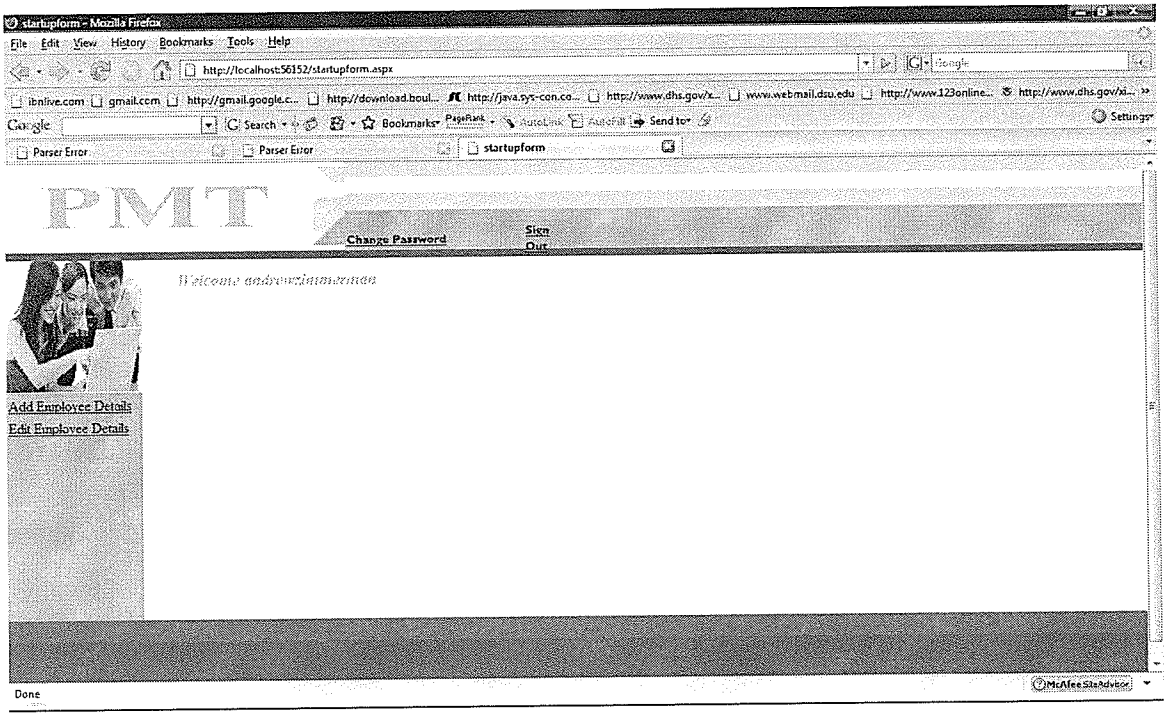

Figure 17. Administrator home page

This is the welcome page for the administrator. Administrator is the person who introduces new employees to the project management tool. He is responsible for maintaining the employee details. He can add or edit the employee details.

Add Employee Details Form

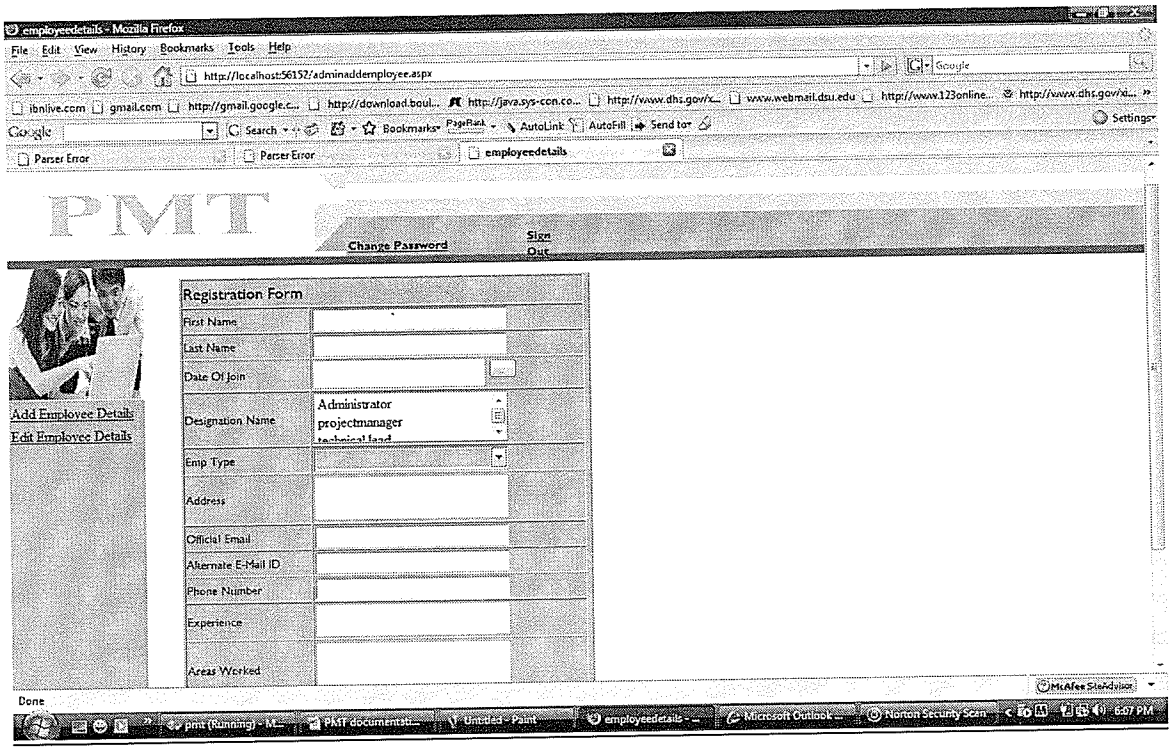

Figure 18. Add Employee Details form

This form is used by Administrator for registering the employees. Whenever a new employee joins the organization administrator enters all the details like employee First name, Last name, Date of joining, Designation, Employee Type, Official E-Mail ID, Alternate E-Mail ID, Employee Phone Number, Experience, Areas Worked etc. After entering the details, the Module will automatically register the employee to the tool. It generates username and password randomly and sends the same to the employee mail box along with URL (uniform resource locator).

Edit Employee Details Form

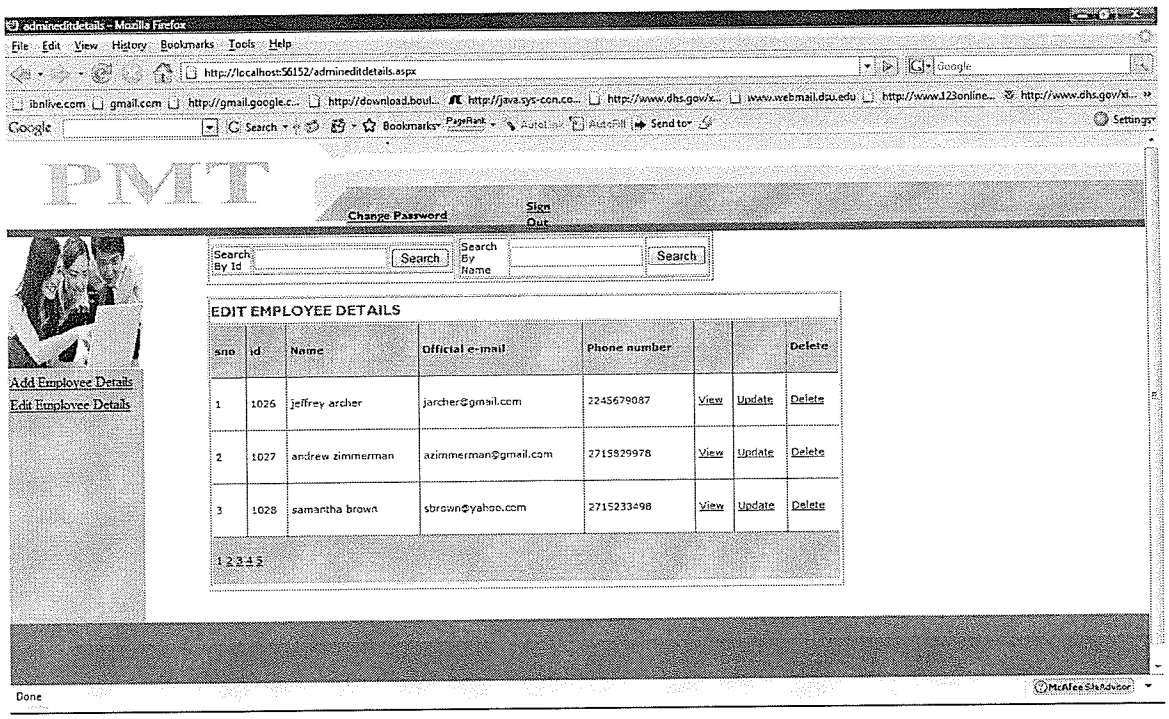

Figure 19. Edit Employee Details form

This form is used by administrator to edit the details of existing employees. Administrator can view the details of the employee. Administrator can update the details of the employee, like a change in Designation, phone number and address etc. Administrator can remove the employees from the tool.

Change Password Form

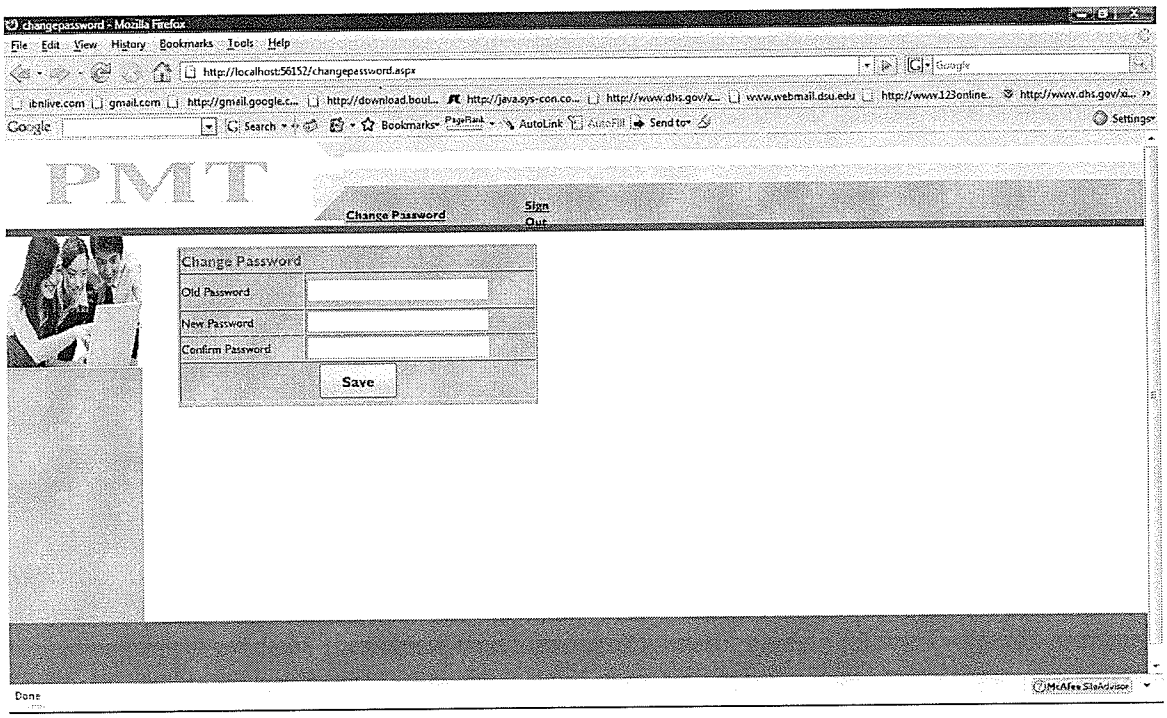

Figure 20. Change password form

This form is used by the employees to change their password. When the employee logs into the PMT for the first time with the user id and password that are randomly generated, he will be given an option to change his password.

Welcome Page For Project Manager

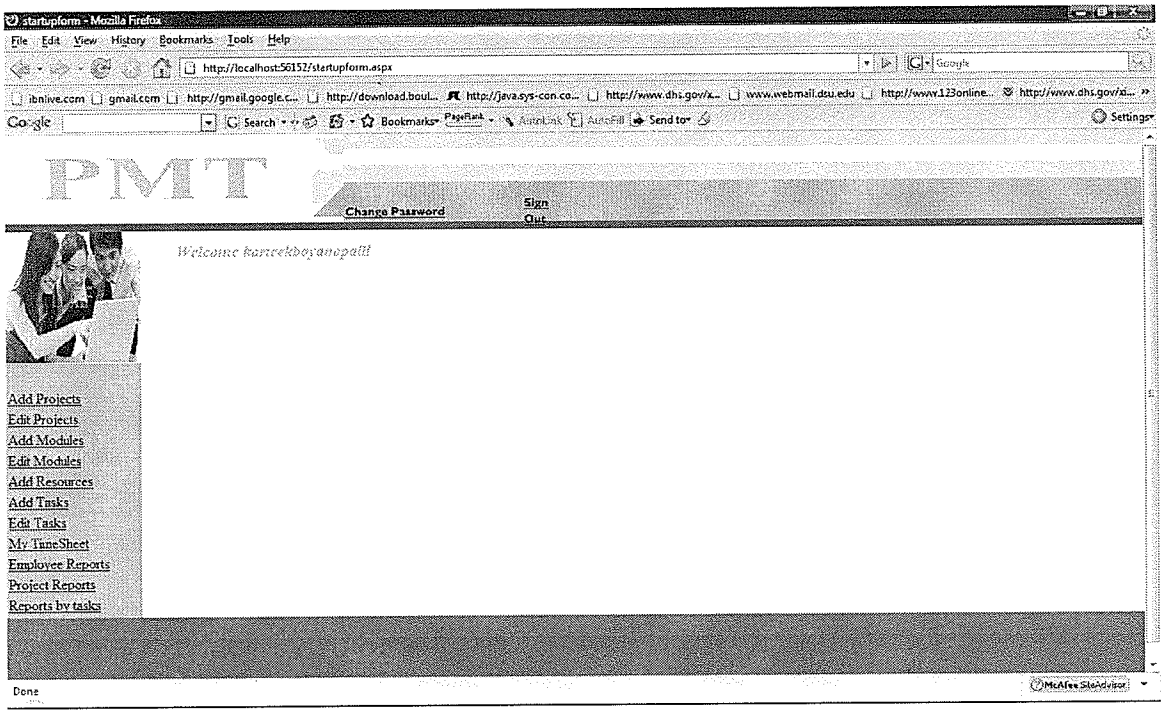

Figure 21. Project Manager home page

This is the welcome page for the project manager .Project manager can add and edit projects, modules and tasks. Project manager can resources for the projects, tasks and modules. He can access different types of reports like, employee reports, project reports and task reports.

Adding Projects Form

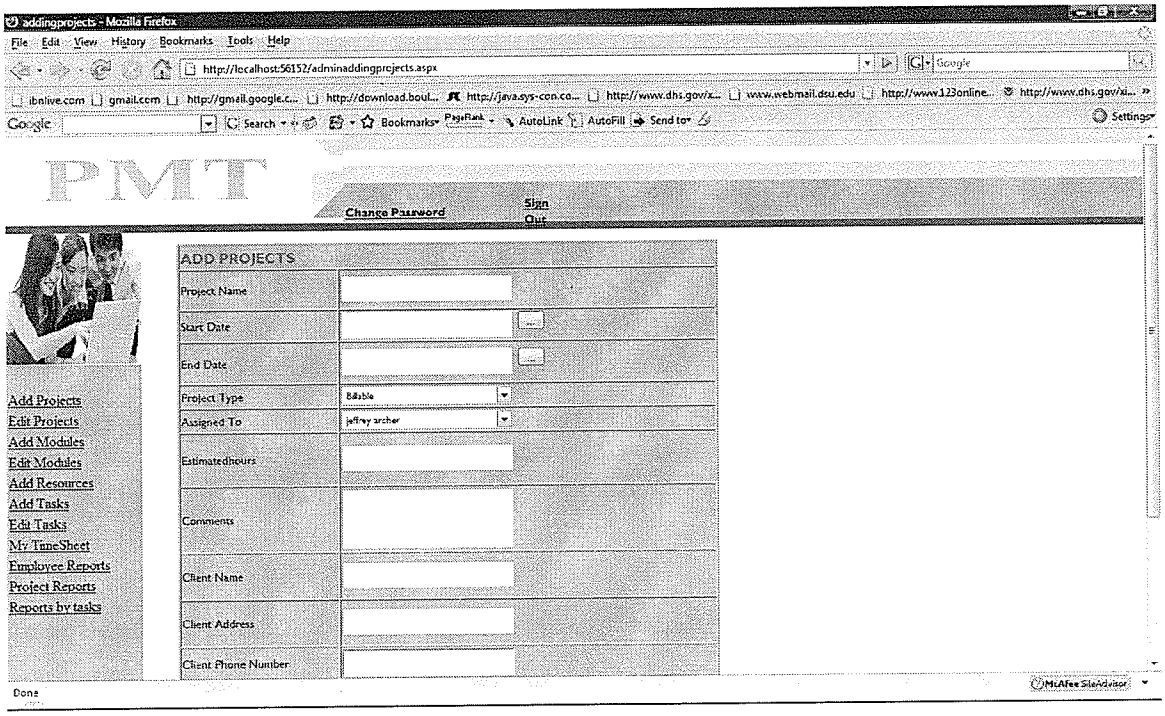

Figure 22. Adding projects form

This form is used by the project manager for adding new projects .After a project is signed by the organization it is assigned to project manager. Project manager estimates the cost and time and adds the projects to the tool.

Project Manager Adding Modules Form

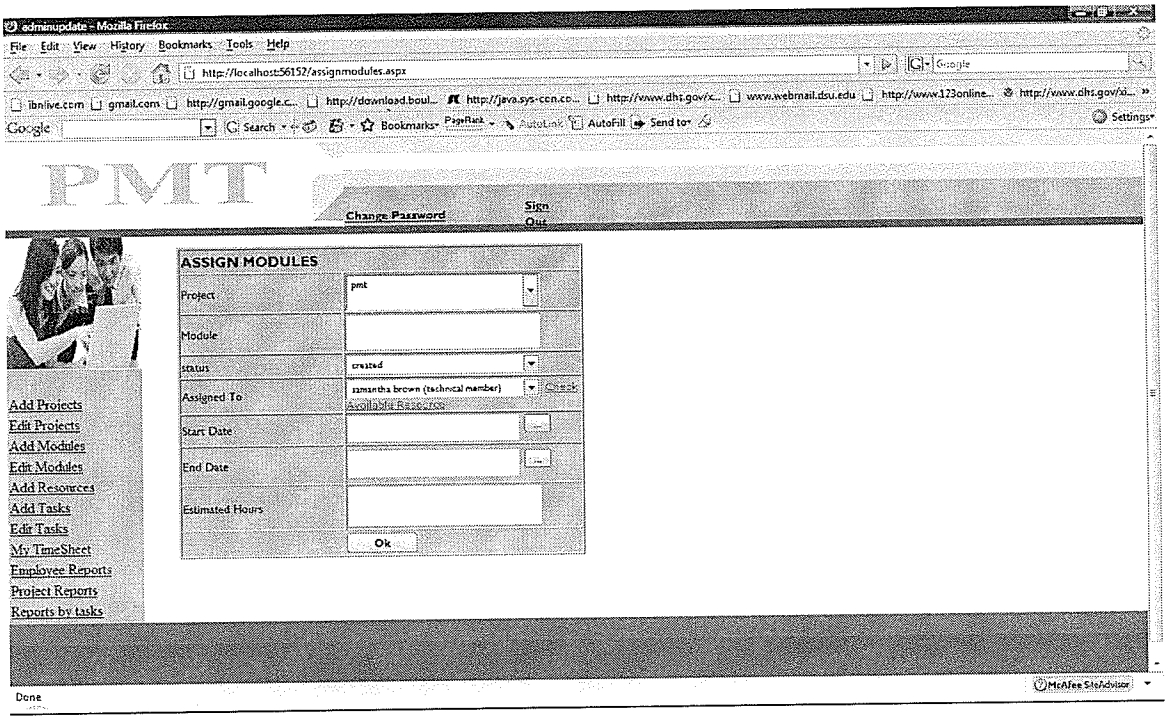

Figure 23. Project Manager Adding Modules form

This form is used by project manager for assigning the modules to team leaders. Generally, Projects are divided into modules. Project manager assigns a module under a project to a particular team leader and keeps track of its status

Project Manager Adding Tasks Form

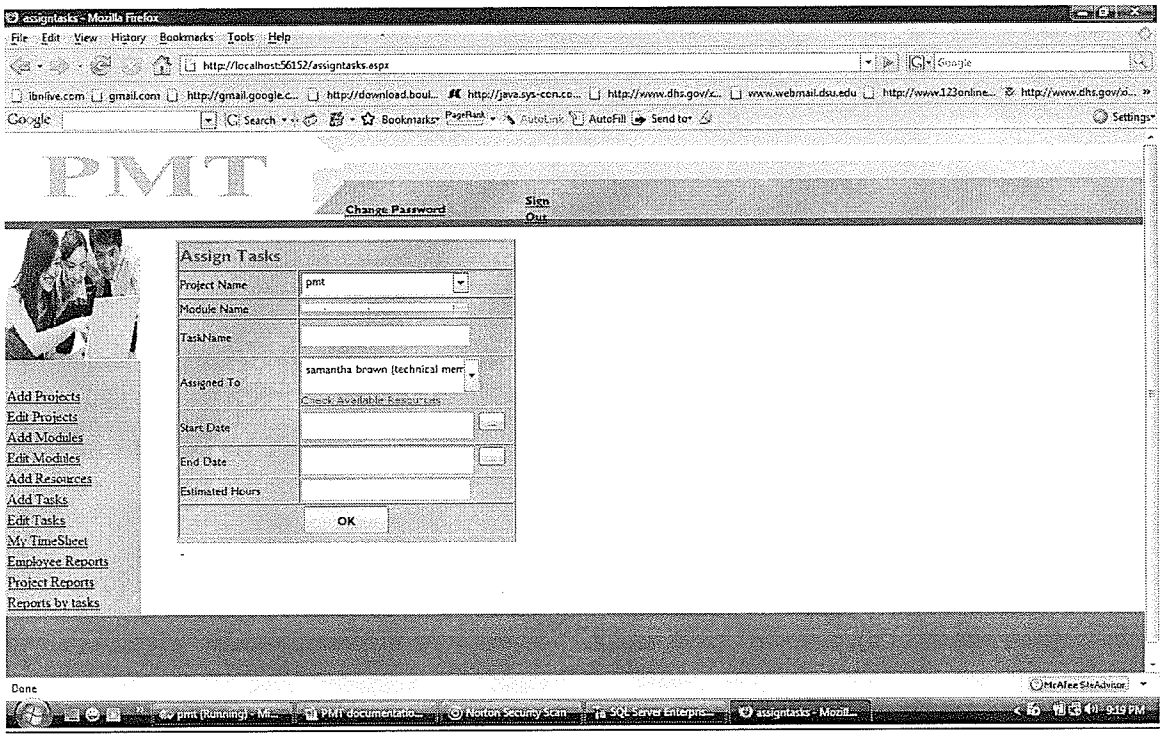

Figure 24. Project Manager Adding Tasks form

For team leader to work on the module he needs group of team members. So, project manager assigns team members to team leaders. Project manager assigns tasks to his subordinates..

Project Reports

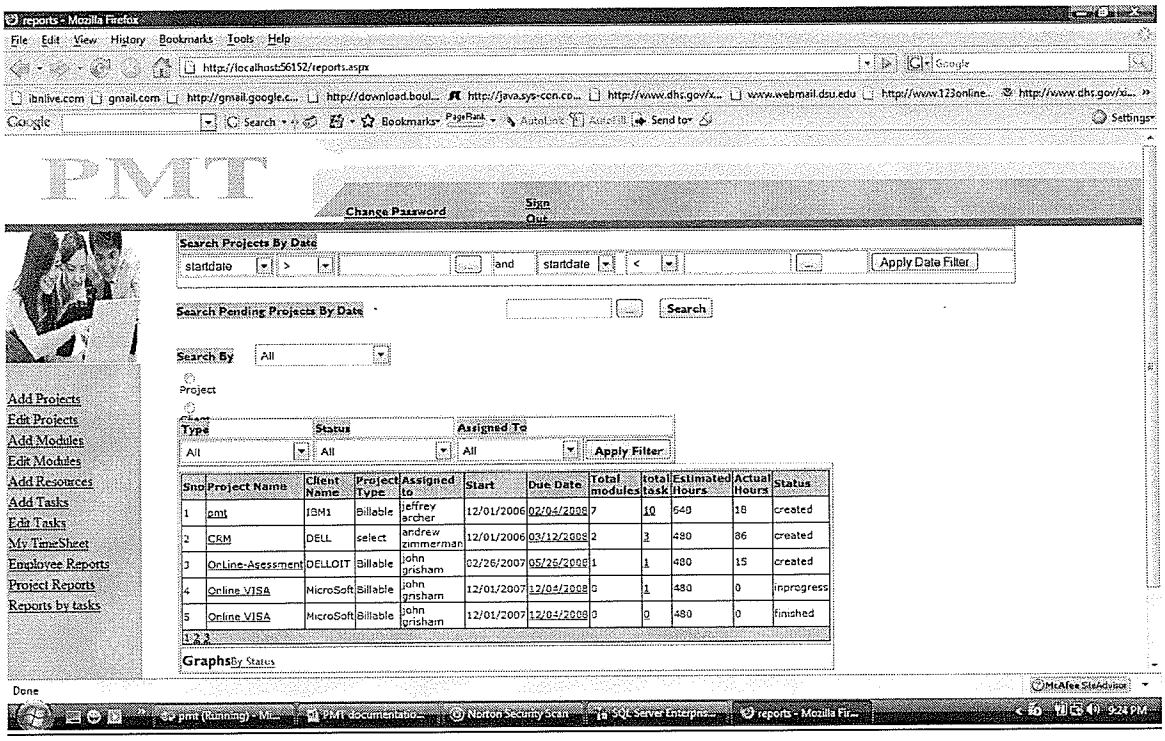

Figure 25. Project Reports form

In the project reports, we can view the details like total no. of modules, tasks, status of the project, number of due date extensions, assigned users. One can also view the detailed description of all the modules and tasks involved in each project.

In addition to above features for the retrieval of the data in the desired fashion we go for filters. The filters provided here are

a) Based on type of the project (Billable/Non Billable).

b) Status of the project.

c) Based on assigned to.

Task Reports

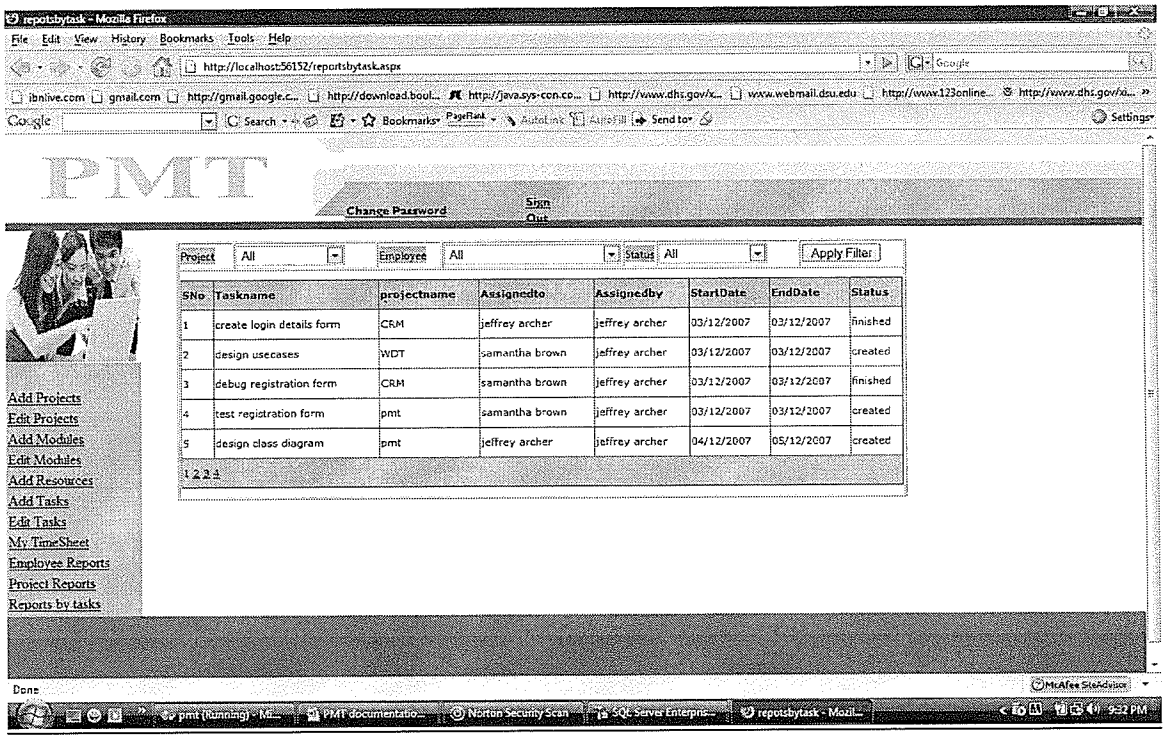

Figure 26. Task Reports form

Task reports give overall tasks carried out in the organization by all employees. Here also we applied filters based on projects, status and assigned to.

Employee Reports

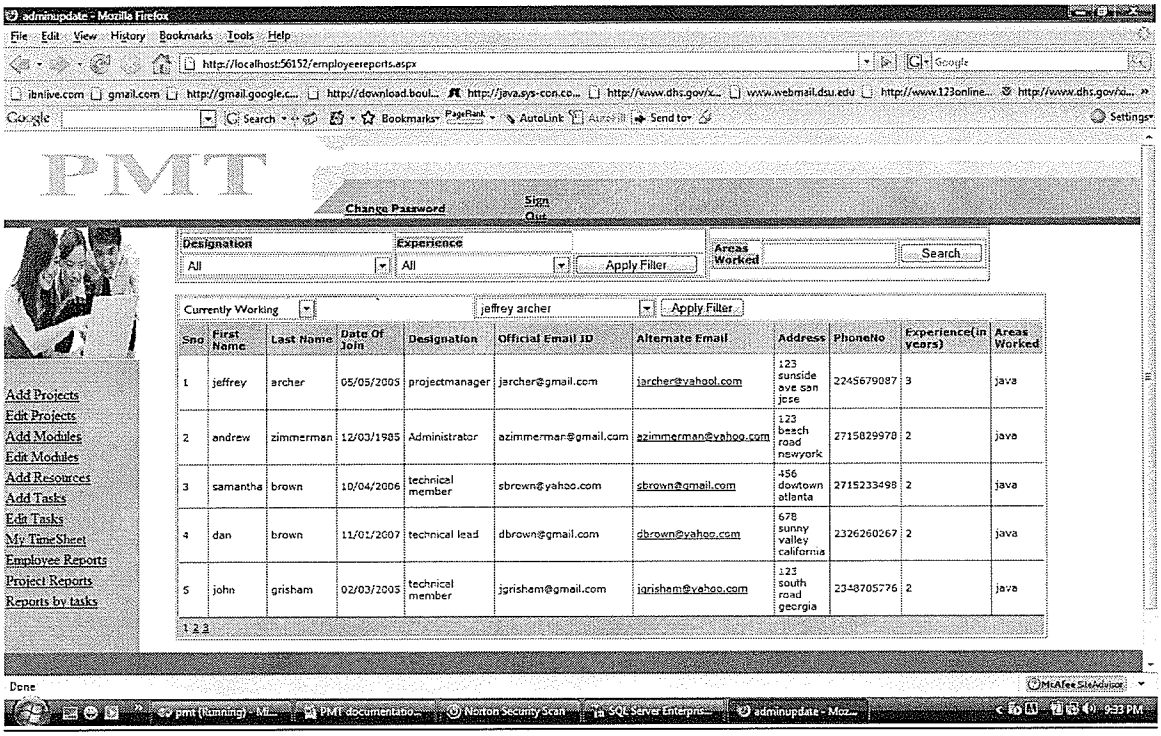

Figure 27. Employee Reports form

Employee reports give the details of all the employees in the organization. i.e currently working and ex-employees.

Project Manager and Administrator can view all the employee details but team leader is restricted to view only his sub-ordinate's details. By seeing the employee reports one can view the total no. of employees.

Employee Reports With Designation As Filter

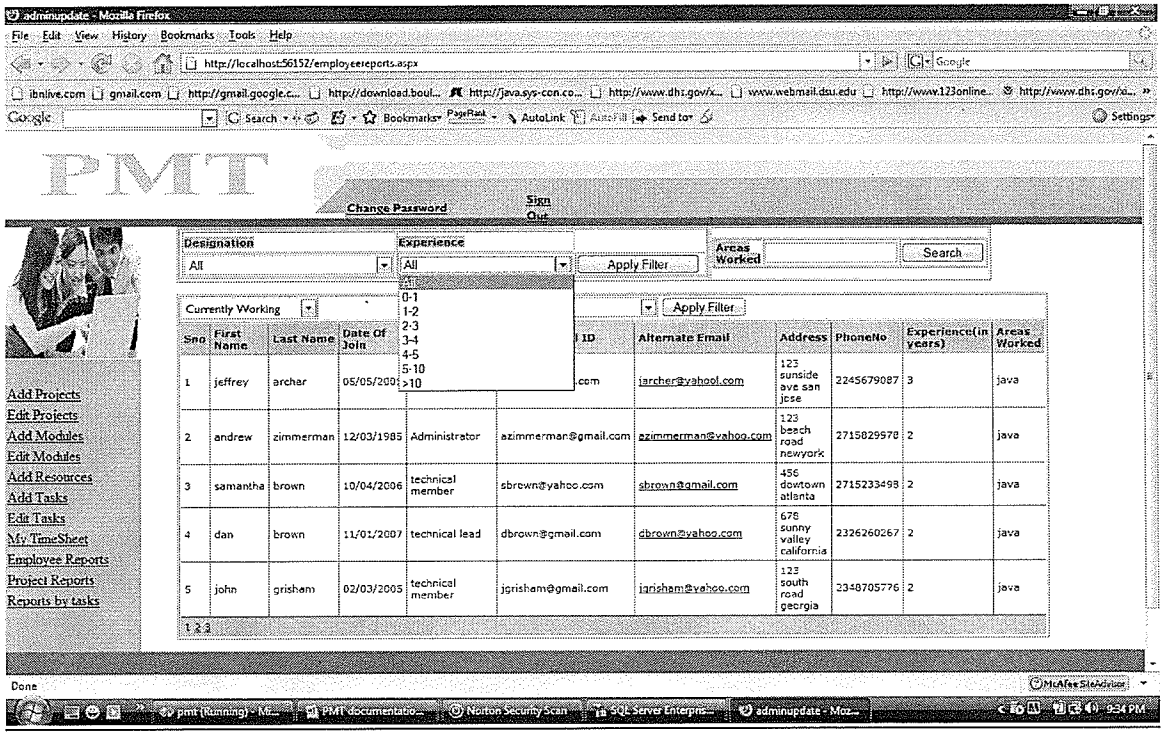

Figure 28. Employee Reports with Designation as filter

To get specific information, here also we can apply filters. The filters provided here are based on designation and experience. Here, we can also search people based on their specialization.

Welcome Page of Team Leader

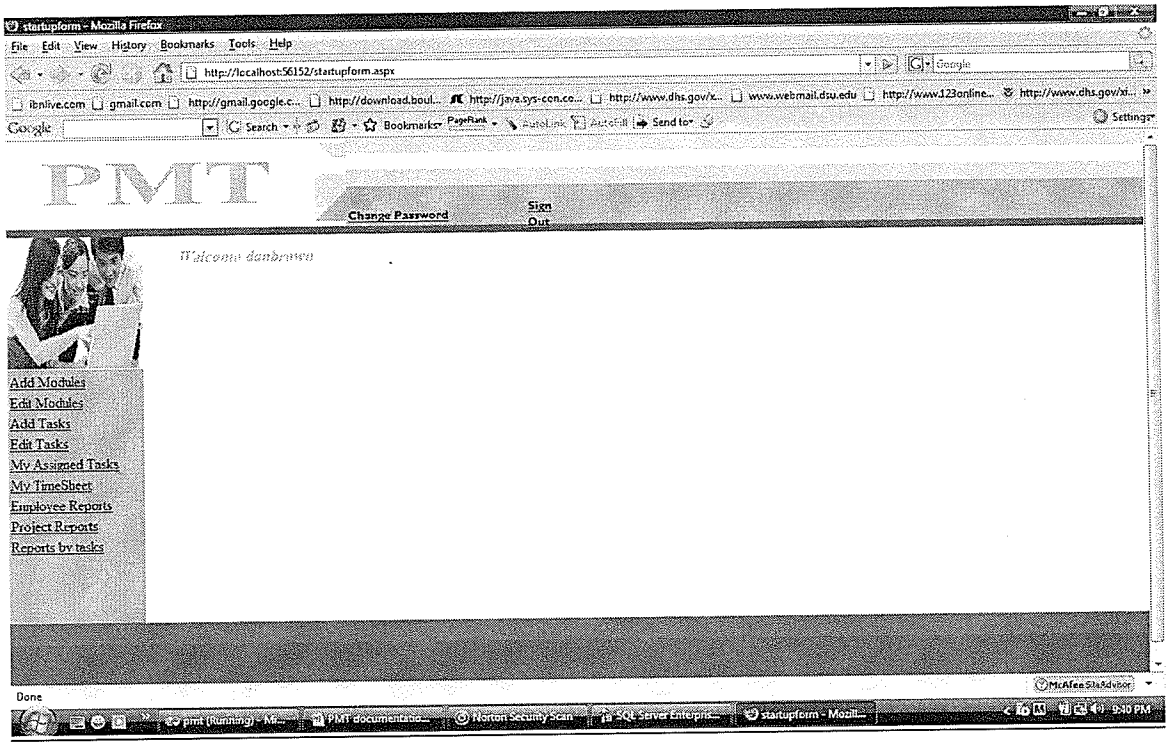

Figure 29. Team leader home page

This is the Welcome page for the Team leader. After the team leader is being assigned a module, he divides the module into tasks, and assigns them to team members. At the end of the day team leader fills timesheet which keeps track of hours worked by him.

Welcome Page of Team Member

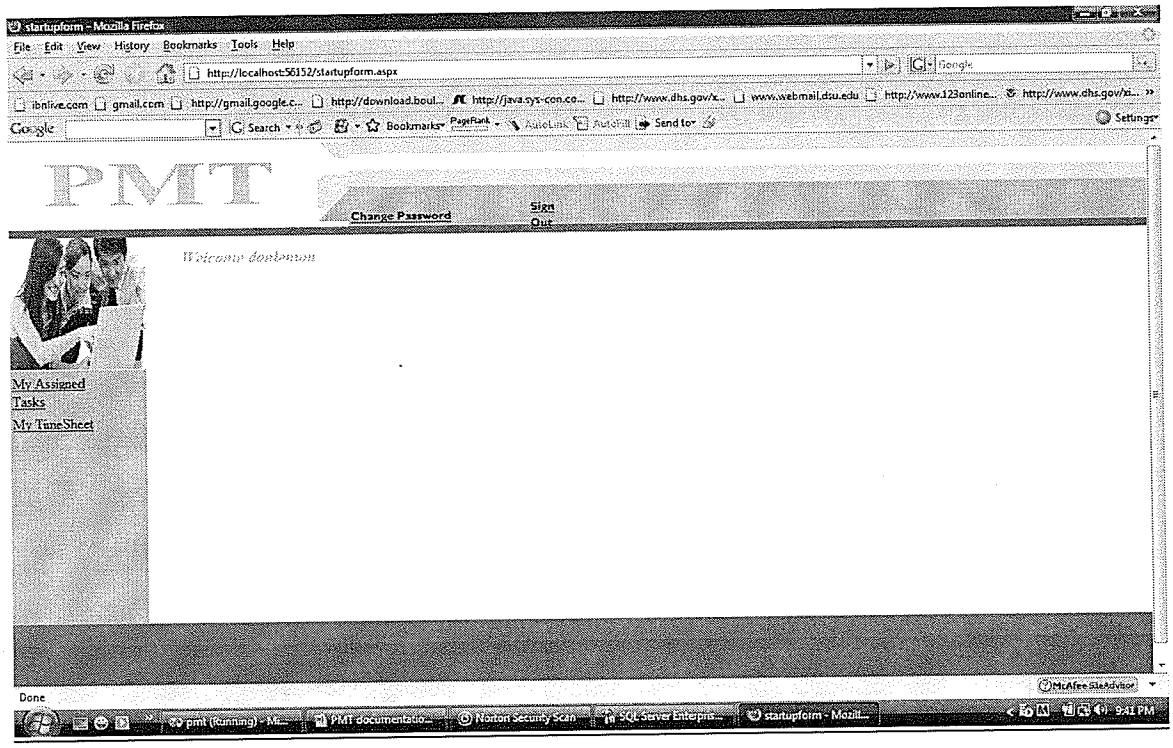

Figure 30. Team Member home page

This is the Welcome page for the Team member. Team member can view all the tasks assigned to him by his superiors and works on those tasks, after the completion of the tasks he submits them by filling time sheet.

CHAPTER 6

TEST CASES

6.1 Test Case for Administrator:

The Administrator login page is created and validated. When the Administrator registers an Employee, a mail is generated to the inbox of the Employee providing the loginid and password .And, it is tested.

6.2 Test Case for Project Manager:

The Project Manager login page is created and validated. Then, the Project Manager can login and view the stored content in the Database through his interface and he can as well add the content to the Database .The Database is created using SqlServer and it is tested.

6.3 Test Case for Team Leader:

The Team Leader login page is created and validated. Then, the Team Leader can login and view the stored content in the Database through his interface and he can as well add the content to the Database. The Database is created using SqlServer and it is tested.

6.4 Test Case for Team Member:

The Team Member login page is created and validated. Then, the Team Member can login and view the stored content in the Database through his interface and he can as well add the content to the Database .The Database is created using SqlServer and it is tested.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

Conclusion:

This Tool can now be used by any organization for effective monitoring of the projects. Using this tool, Project Manager and Team Leader can delegate tasks to their subordinates. The actual number of working hours of the Employees is tracked effectively through Timesheets and the Required Status can be viewed by the Project, Employee and Task by Reports.

Future Enhancements:

1. Discussion Forum: This feature helps to carry out effective communication between employees.
2. Message Board: This feature enables an employee to post any important article so that it can be viewed by all the employees in the organization.

REFERENCES

Textbooks

1) Matthew MacDonald, ASP.net Complete Reference, Tata McGraw-Hill publications, 2002

2) G.Andrew Duthie,Microsoft Asp.net,Step by Step, Eastern Economy, Tata McGraw-Hill Publishing Company,Hyderabad,2002..

Reference books

GregBuczek,Mcsp,MCT, ASP Developers Guide, Tata McGraw-Hill Publishing Company,Hyderabad,2002..

Websites

http://www.aspdeveloper.net/tiki-index.php

http://www.startvbdotnet.com/

http://programmerworld.net/dotnet/books.htm

APPENDICES

APPENDIX-1

Dot Net

.NET is the Microsoft Web services strategy to connect information, people, systems, and devices through software. Integrated across the Microsoft platform, .NET technology provides the ability to quickly build, deploy, manage, and use connected, security-enhanced solutions with Web services. .NET connected solutions enable businesses to integrate their systems more rapidly and in a more agile manner and help them realize the promise of information anytime, anywhere, on any device.

.Net is basically developed to support language independent applications..Net programs are independent of Operating System and Hardware.Like Java has JVM, .Net also includes Virtual Machine known as C.L.R (Common Language Runtime).

Traditional way of executing programs:

The source code and pre compiled reference libraries are given to compiler and they are compiled and generating an exe file which is platform dependent.

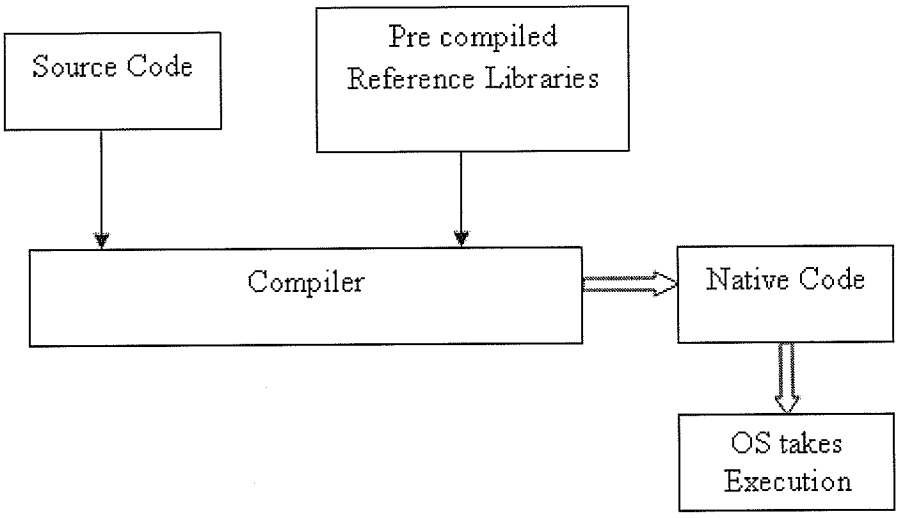

Figure 31. Traditional way of executing programs

.net way of executing programs

Source code and uncompiled reference libraries are given to the compiler and the file is generated after compiling is of the form M.S.I.L(MicroSoft Intermediate Language).

The CLR(Common Language Runtime) converts this code into executable at the runtime.

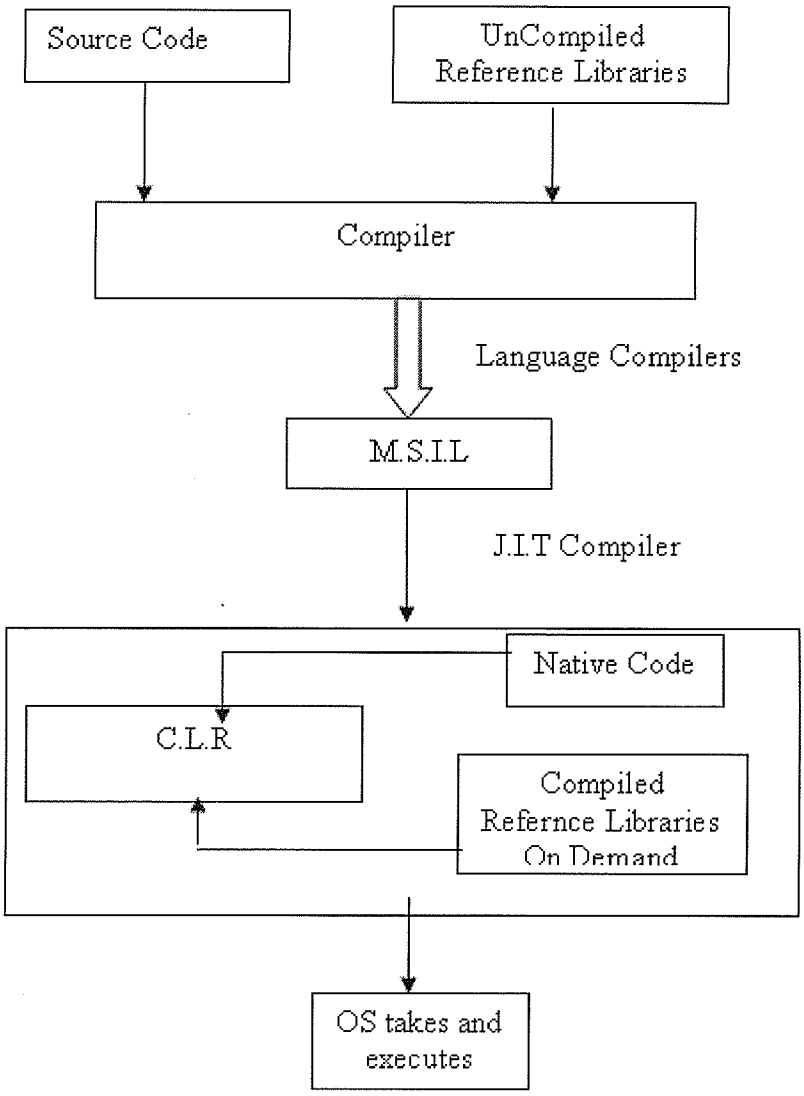

Figure 32. Dot Net way of executing programs

.NET Framework

.Net Framework is an environment for building, deploying and running Web services and other applications.

.Net Framework components are:

- C.L.R

- M.S.I.L
- JITters
- Garbage Collector
- Base Class Libraries
- Manage Code VS Unmanaged Code
- Common Language Specification(CLS)
- Common Type System(CTS)

C.L.R (Common Language Runtime)

It is a collection of resources which handles the execution of .NET programs as we have JVM to execute java programs.

M.S.I.L (Microsoft Intermediate Language)

When we compile .Net compliant languages, the file which is generated is of the form M.S.I.L. This file is platform or CPU independent. While executing the MSIL code is converted into native code which is platform dependent or CPU dependent. While executing these programs CLR activates JITers(JIT compilers). One more file generated is called metadata which gives information about he program and reference libraries.

JITers(Just In Time Compilers)

When interpreted language compiled code need to be executed, CLR invokes JIT compiler, which compile Intermediate Language Code to Native code for specific machine or operating system.

There are 2 types of JITers

1) Standard JIT
2) Econo JIT

Standard JIT:

Entire code assembly is taken and converted to native code with respect to operating system. The result is stored in cache. For further requests, response will be given from cache.

Econo JIT:

Requested contents are converted to native code on demand. Results are not stored in cache.

To execute .Net programs under UNIX platform we need MONO JIT compiler.

Garbage Collector:

Garbage collector deallocates memory of variables which are not in use. CLR also contains garbage collector which runs in lower priority thread. Garbage collector is called only when application memory is full.

Base Class Libraries:

.Net provides base class library for common and usual tasks. To use any class we need to add a reference.

Base class libraries:

System

System. data

System. drawing

System.xml

System.windows.forms

Manage Code VS Unmanaged Code

If the programs executed under the control of CLR they are said to be managed code.

Code that doesn't execute within the C.L.R. is called Unmanaged Code.

CLS (Common Language Specification)

Microsoft released a specification that each language need to qualify then considered as .Net Compliant Language.

If your code is in CLS boundary i.e., CLS compliant then it can be used by other .Net Languages.

Common Type System:

C.T.S defines the data types that intermediate language understands.

Features of .Net:

- Language Independence
- Platform Independence
- Hardware Independence
- ADO.net

.Net also allows users to connect to databases in 2 ways:

Connection oriented: Here, the connection is required in between application and database continuously.

Disconnected Model: Connection is required between application and database only when executing queries.

- Security

  .Net provides security for windows applications as well as web applications.

  Windows application: Code Access Security and Role Based Security

  Web Application: Form Based Security, Windows Security and Password Authentication.

- Globalization and localization
- Language interoperability

APPENDIX-2

Microsoft Sql Server

SQL Server is a relational database management system produced by Microsoft. It supports a superset of Structured Query Language SQL, the most common database language. **SQL Server** is easy to manage and provides a separate OLAP engine. It is the ideal database for web applications written in Asp/Asp.Net.

Features of SQL Server 2000

1) Internet Integration.

The SQL Server 2000 database engine includes integrated XML support. It also has the scalability, availability, and security features required to operate as the data storage component of the largest Web sites. SQL Server 2000 supports features such as English Query and the Microsoft Search Service to incorporate user-friendly queries and powerful search capabilities in Web applications.

2) Scalability and Availability.

The same database engine can be used across platforms ranging from laptop to multiprocessor servers. SQL Server 2000 Enterprise Edition supports features such as indexed views, and large memory support that allow it to scale to the performance levels required by the largest Web sites.

3) Enterprise-Level Database Features.

The SQL Server 2000 relational database engine supports the features required to support demanding data processing environments. The database engine protects data integrity while minimizing the overhead of managing thousands of users concurrently modifying the database. SQL Server 2000 distributed queries allow you to reference data from multiple sources as if it were a part of a SQL Server 2000 database, while at

the same time, the distributed transaction support protects the integrity of any updates of the distributed data. Replication allows you to maintain multiple copies of data, while ensuring that the separate copies remain synchronized. You can replicate a set of data to multiple, mobile, disconnected users, have them work autonomously, and then merge their modifications back to the publisher.

4) Ease of installation, deployment, and use.

SQL Server 2000 includes a set of administrative and development tools that improve upon the process of installing, deploying, managing, and using SQL Server across several sites. .

Database Architecture

Microsoft SQL Server 2000 data is stored in databases. The data in a database is organized into the logical components visible to users. A database is also physically implemented as two or more files on disk.

When using a database, we work primarily with the logical components such as tables, views, procedures, and users. The physical implementation of files is largely transparent. Typically, only the database administrator needs to work with the physical implementation.

Each instance of SQL Server has four system databases (master, model, tempdb, and msdb) and one or more user databases.

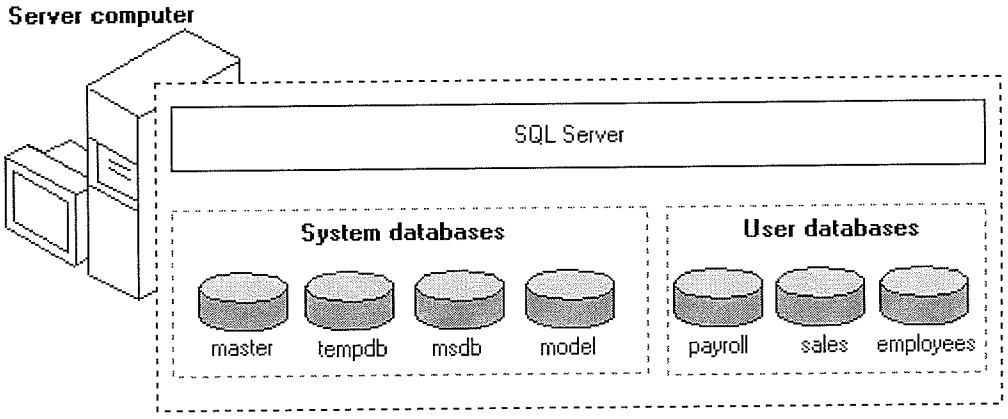

Figure 33.SQL Server

SQL Database Tables are the foundation of every RDBMS (Relational Database Management System). To retrieve data from tables sql commands are used.

Several SQL commands:

SQLSELECT

SELECT SQL statement is used to retrieve data from a SQL database table.

SQL SELECT INTO

SQL SELECT INTO statement is used to copy data between database tables.

SQL DISTINCT

SQL DISTINCT clause together with the SQL SELECT keyword is used to return a dataset with unique entries for certain database table column.

SQL WHERE

The SQL WHERE command is used to specify selection criteria, thus restricting the result of a SQL query.

SQL LIKE

The SQL LIKE clause is used along with the SQL WHERE clause and specifies criteria based on a string pattern.

SQL INSERT INTO

SQL INSERT INTO clause is used to insert data into a SQL database table.

SQL UPDATE

SQL UPDATE statement is used to update data in a SQL database table.

SQL DELETE

SQL DELETE statement is used to delete data from a SQL database table.

SQL ORDER BY

SQL ORDER BY statement is used to sort the data retrieved in your SQL query.

SQL OR & AND

SQL OR & AND keywords together with the SQL WHERE clause is used to add several conditions to your SQL statement.

SQL IN

The SQL IN clause allows you to specify discrete values in your SQL WHERE search criteria.

SQL BETWEEN

The SQL BETWEEN & AND keywords define a range of data between 2 values.

SQL Aliases

SQL aliases can be used with database tables and/or with database table columns, depending on task you are performing.

SQL COUNT

SQL COUNT aggregate function is used to count the number of rows in a database table.

SQL MAX

SQL MAX aggregate function allows us to select the highest (maximum) value for a certain column.

SQL MIN

SQL MIN aggregate function allows us to select the lowest (minimum) value for a certain column.

SQL AVG

The SQL AVG aggregate function selects the average value for a certain table column.

SQL SUM

SQL SUM aggregate function allows selecting the total for a numeric column.

APPENDIX-3

Screens

View Employee Details

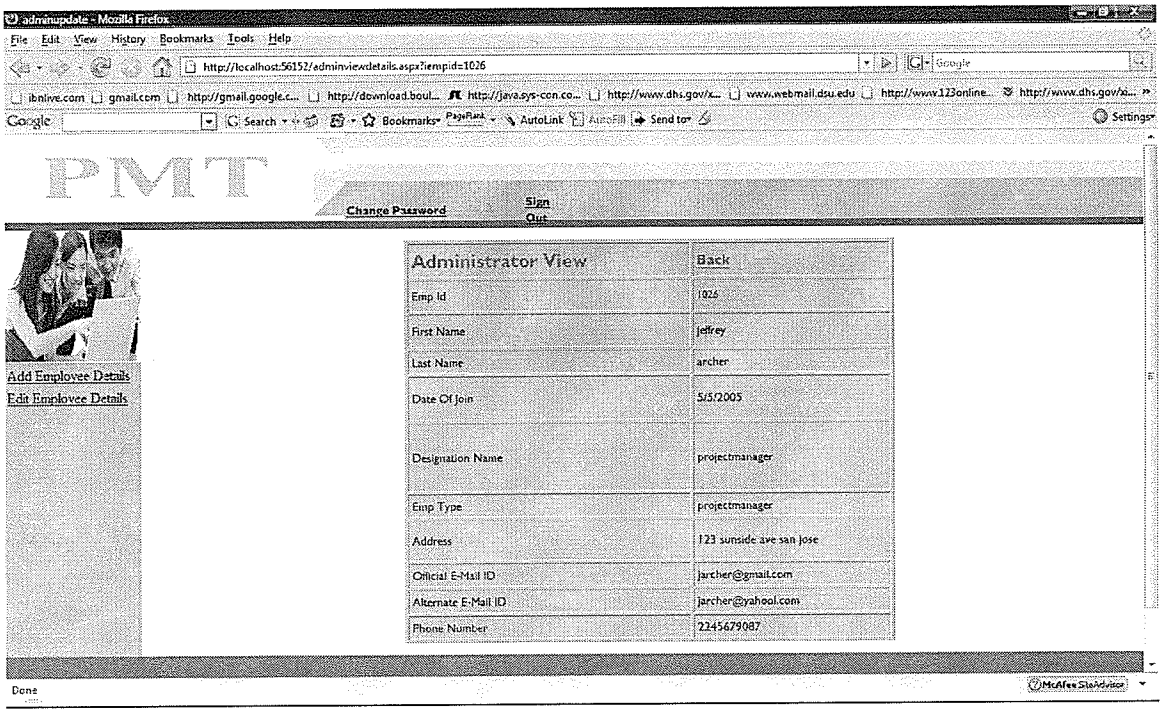

Figure 34. Employee view details

Update Employee Details Form

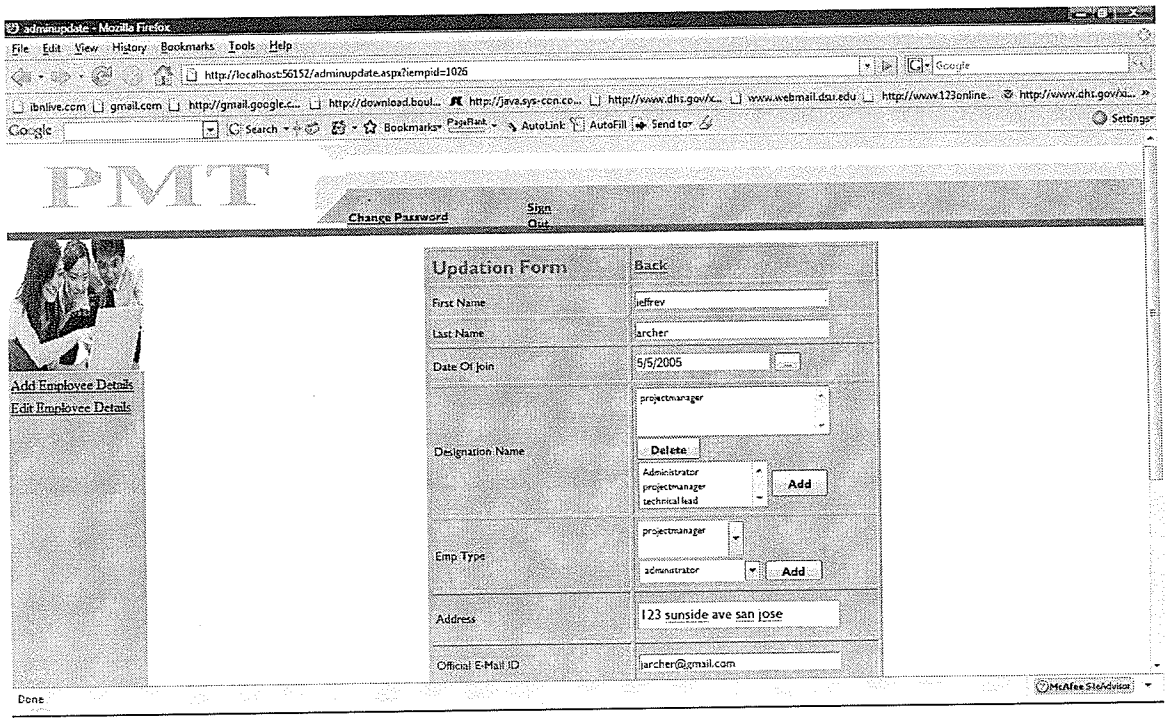

Figure 35. Update employee details form

Editing Project Details Form

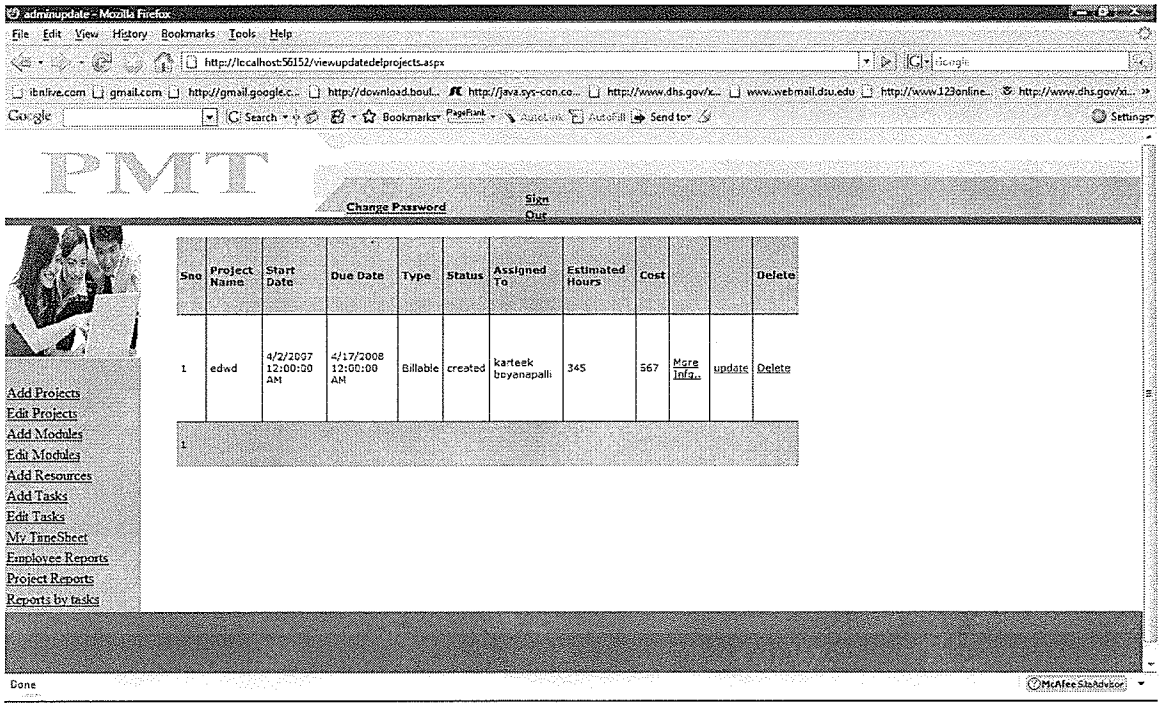

Figure 36. Editing project details form

Project Manager: Updating Project Details

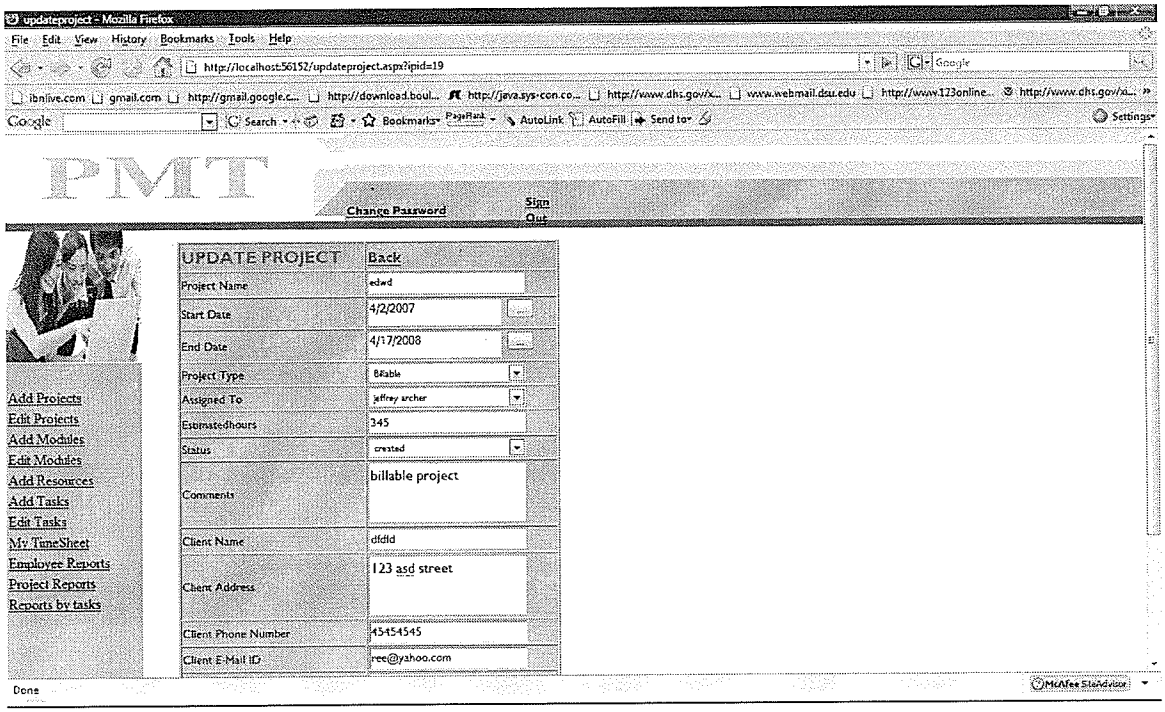

Figure 37. Project Manager Update Project Details form

Project Manager: Deleting Project

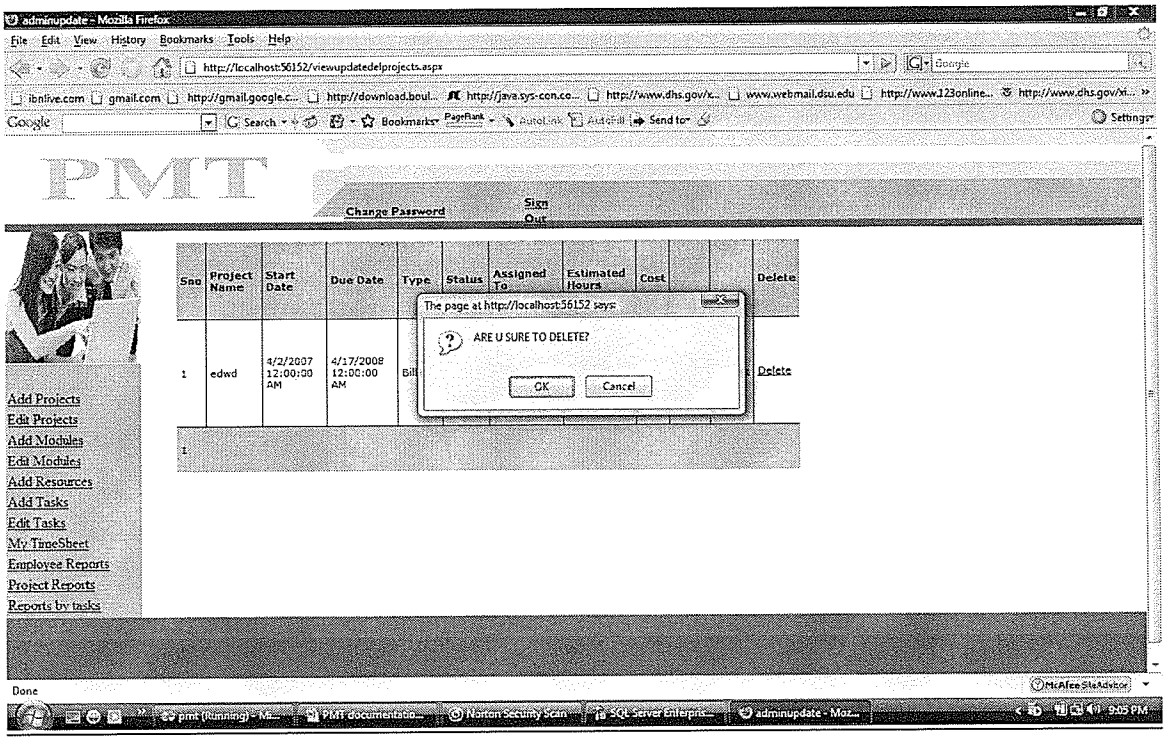

Figure 38. Project manager deleting project

Checking Available Resources

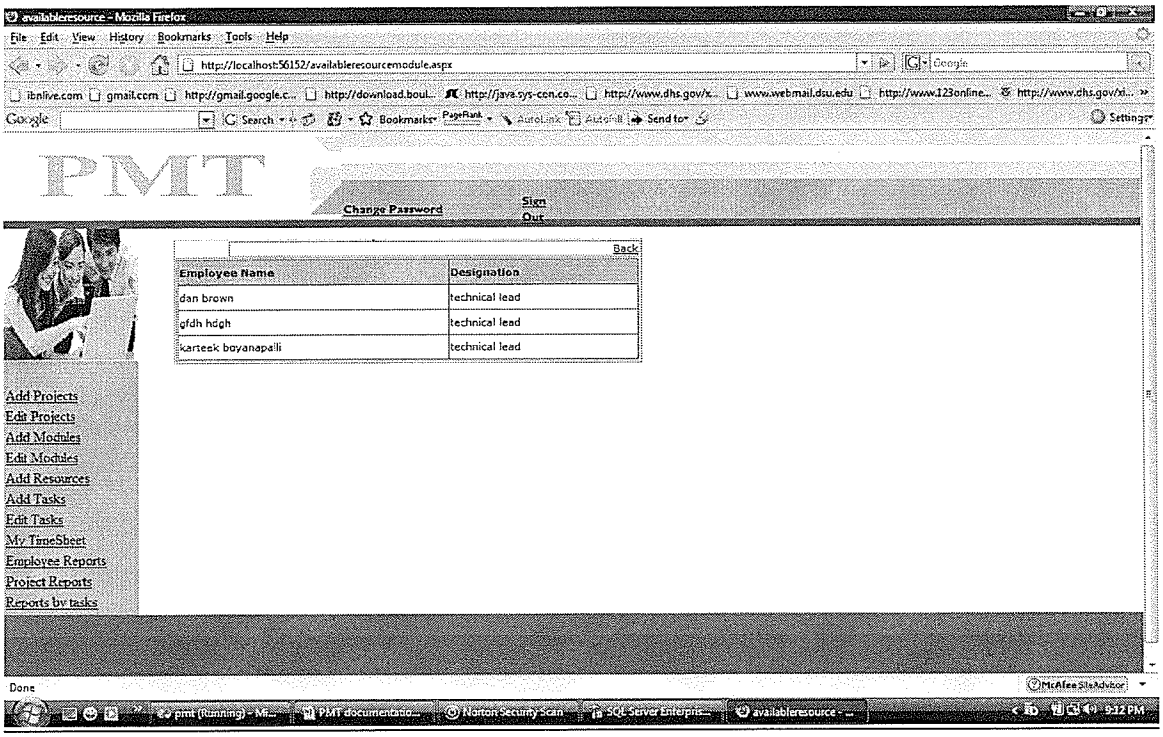

Figure 39. Check available resources

Project Manager Editing Modules Form

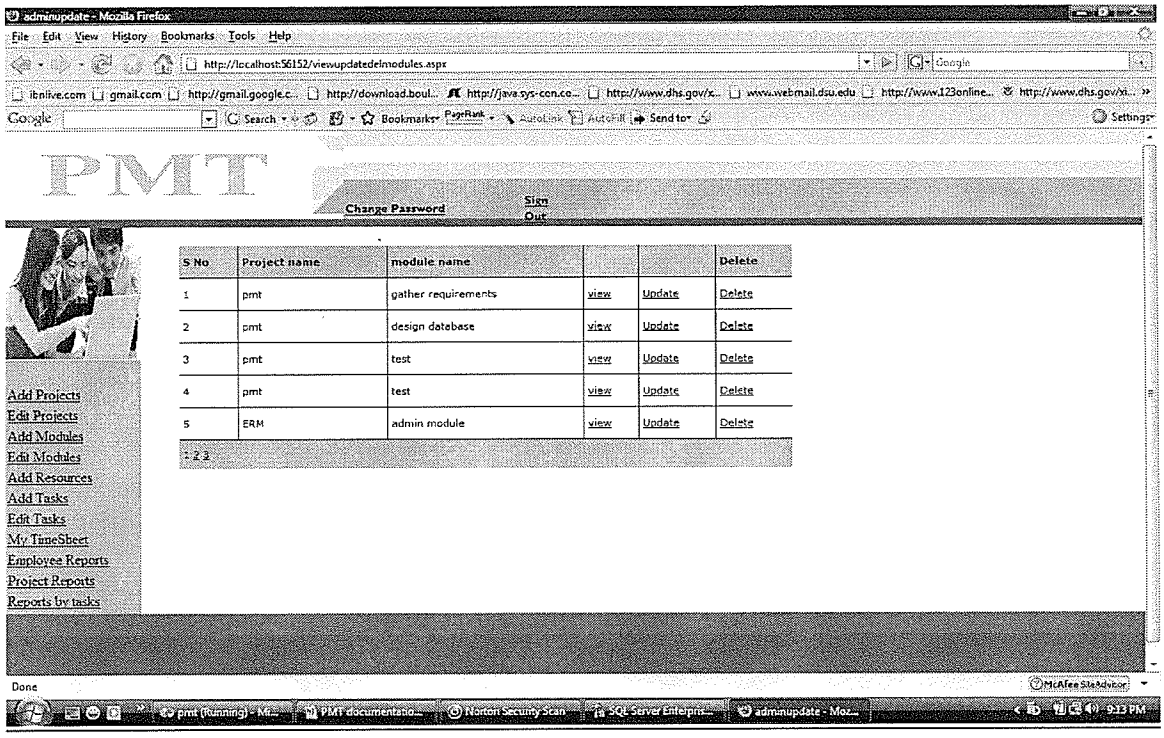

Figure 40. Project Manager Editing Modules form

Project Manager Viewing Modules Form

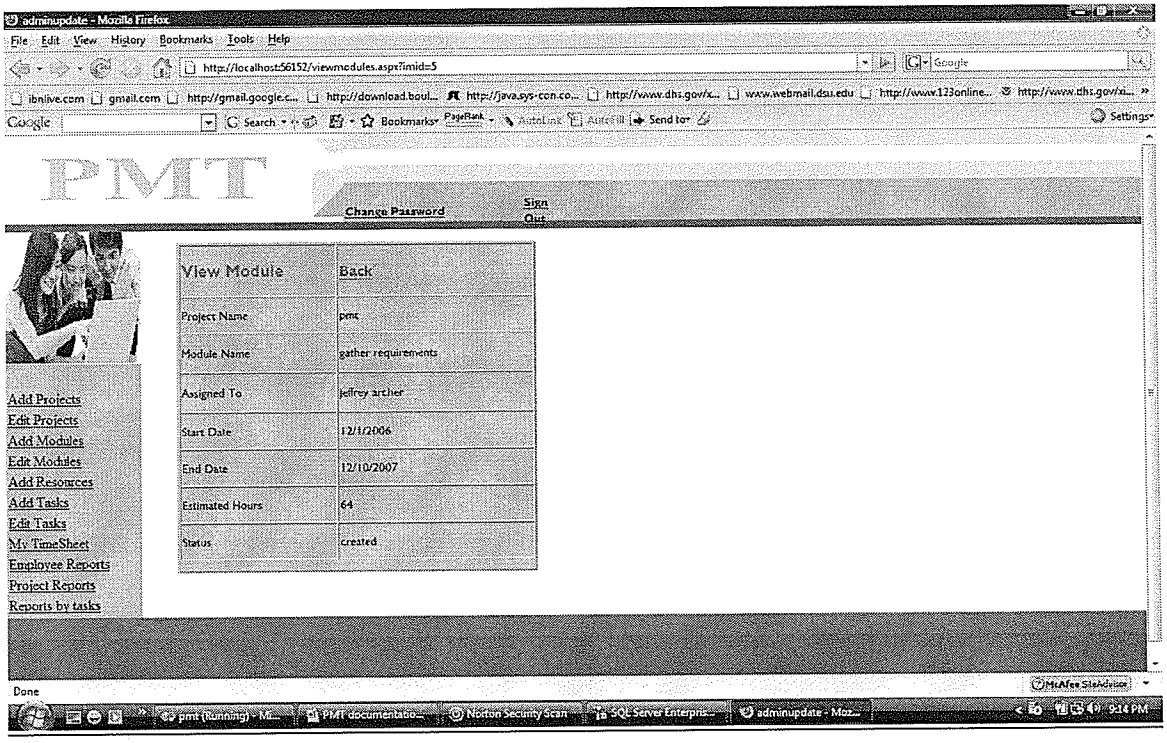

Figure 41. Project Manager Viewing Modules form

Project Manager Updating Modules Form

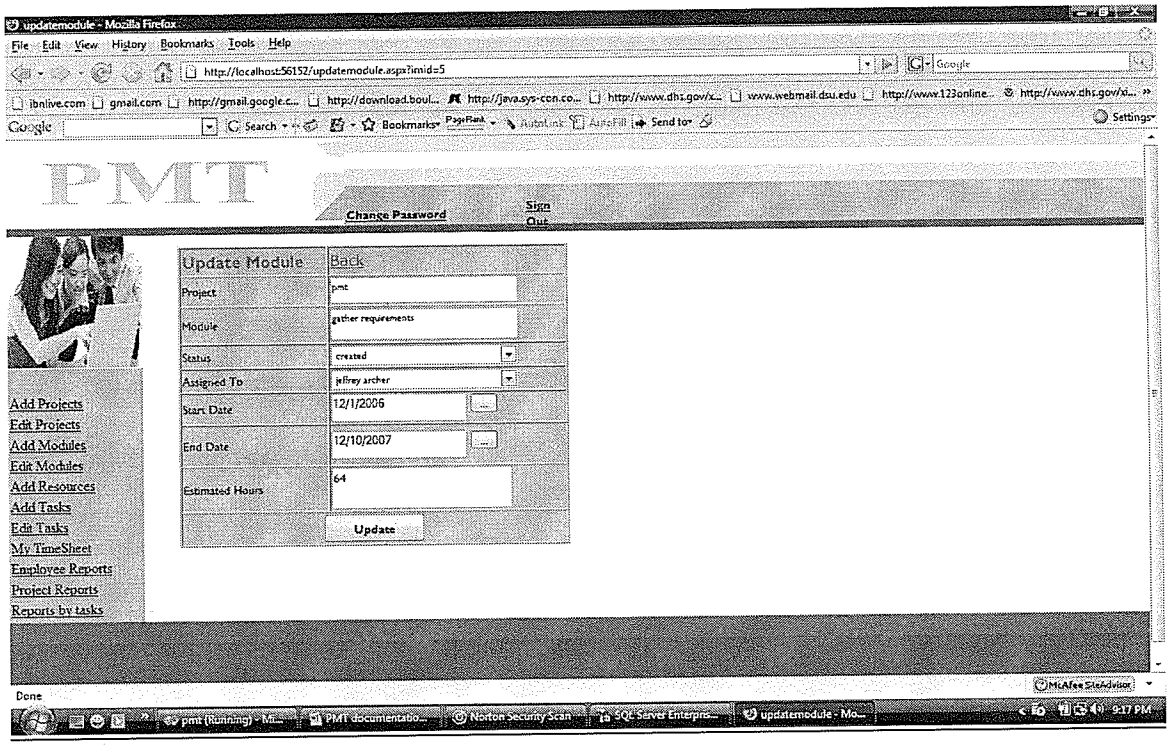

Figure 42. Project Manager Updating Modules form

Project Manager Adding Resources Form

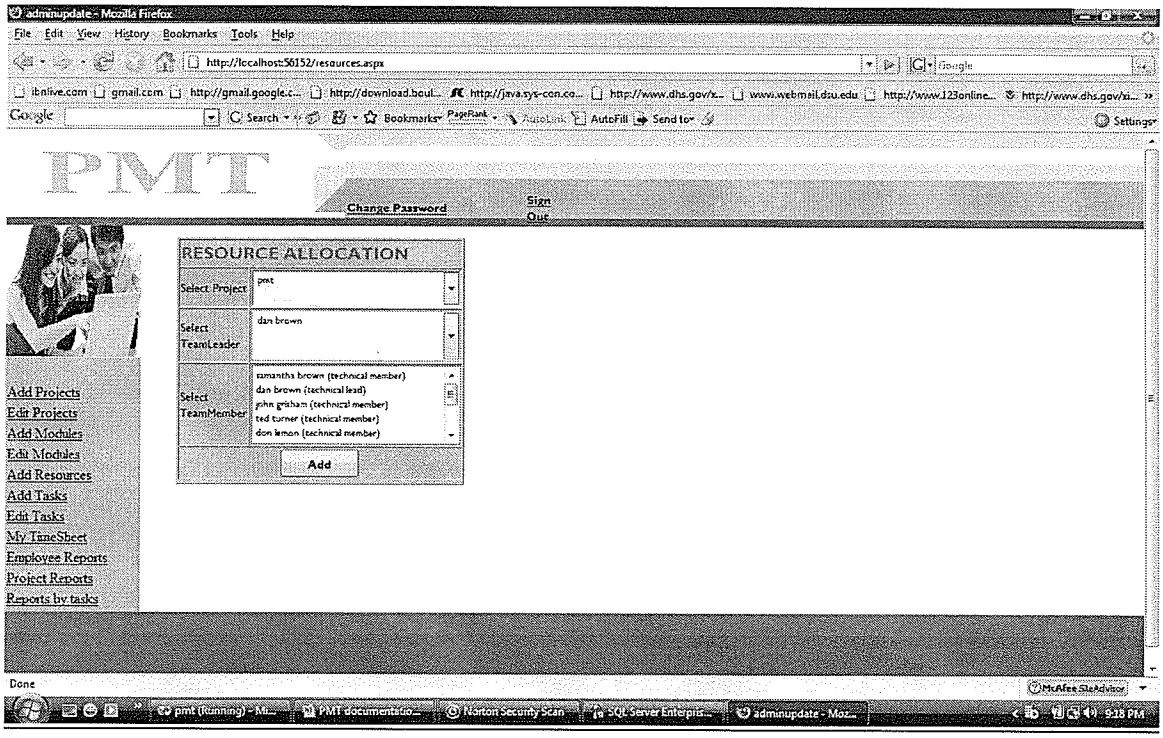

Figure 43. Project Manager Adding Resources form

Project Manager Editing Tasks Form

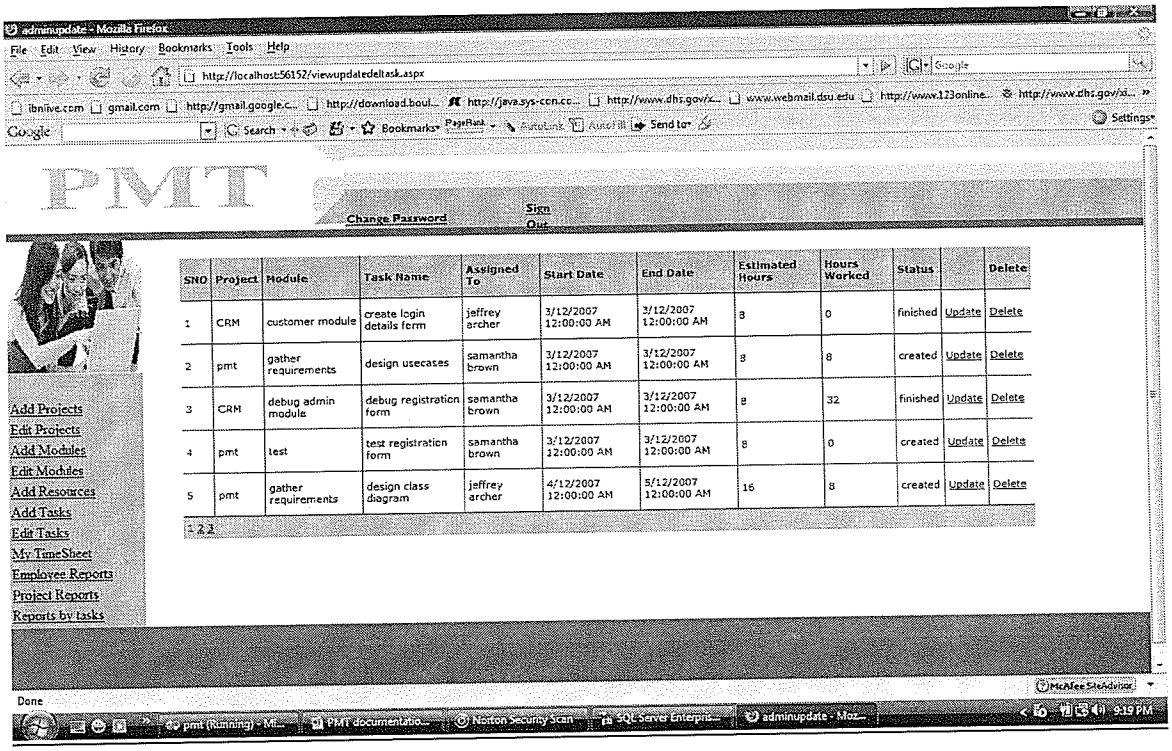

Figure 44. Project Manager Editing Tasks form

Project Manager Updating Task Form:

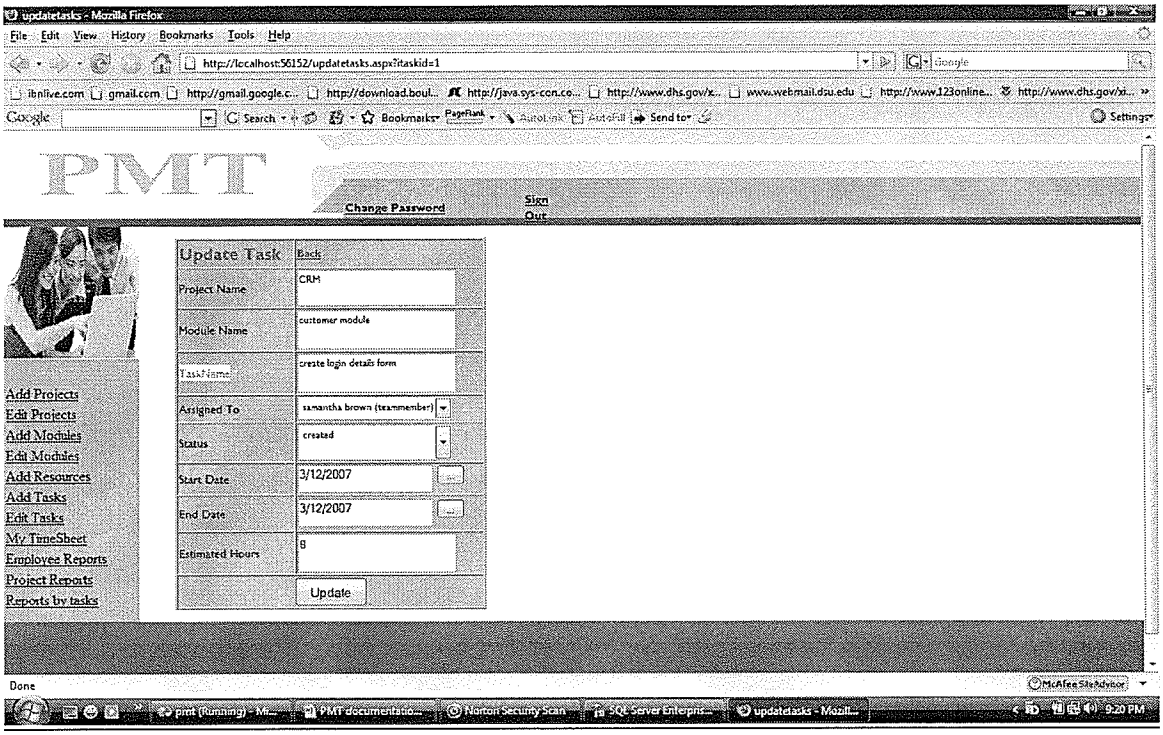

Figure 45. Project Manager Updating Tasks form

Viewing Task Details From Project Reports

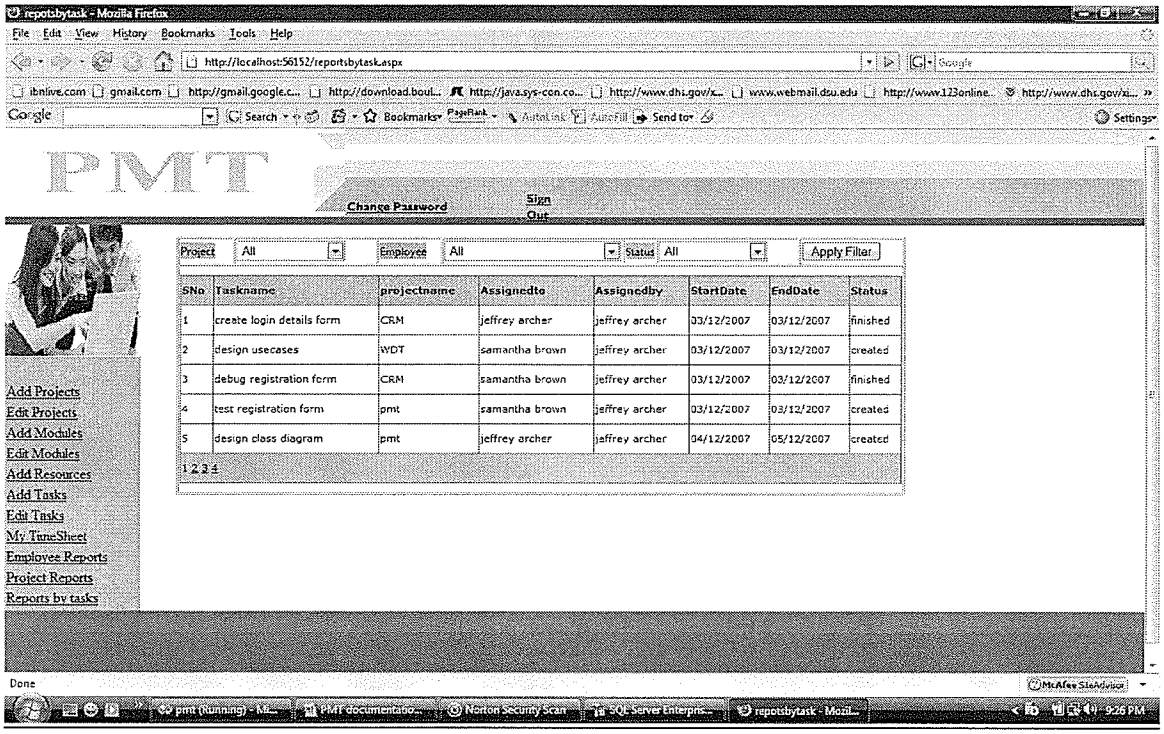

Figure 46. Viewing task details form project reports

Form To Send E-Mails

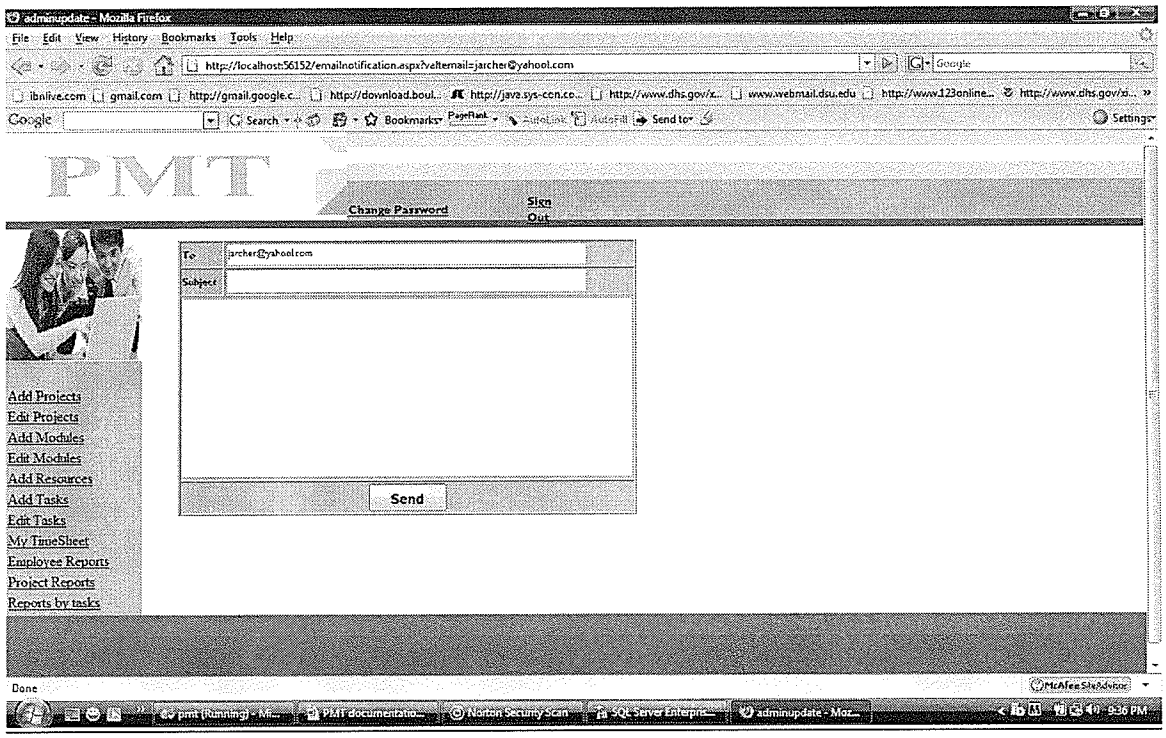

Figure 47. Send Emails form

Team member checking His Assigned Tasks

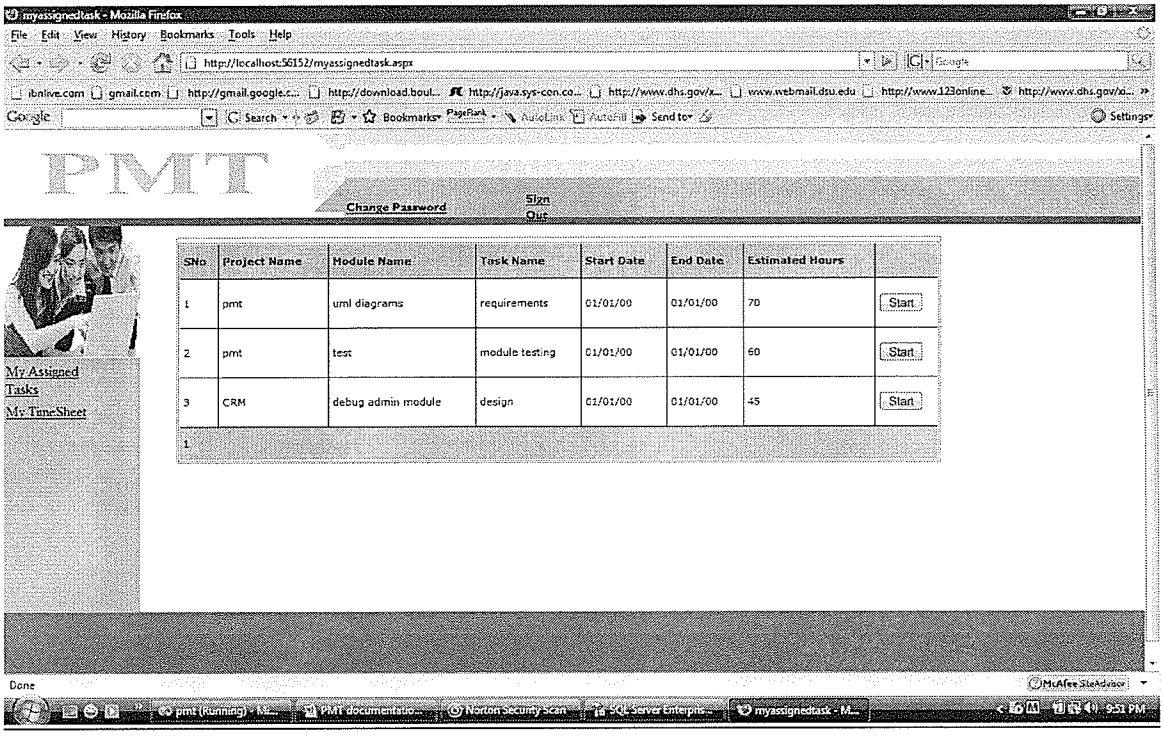

Figure 48. Team Member checking his assigned tasks